

UNIVERSITÄT
BAYREUTH

Intuitive Roboterkommandierung basierend auf Verbalisierten Physikalischen Effekten

Von der Universität Bayreuth
zur Erlangung des Grades eines
Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigte Abhandlung

von
Michael Spangenberg
aus Auerbach/Opf.

1. Gutachter: Prof. Dr. Dominik Henrich
2. Gutachter: Prof. Dr. Diedrich Wolter

Tag der Einreichung: 03.03.2017
Tag des Kolloquiums: 09.06.2017

Abstract

One long term goal in current robotics research is the development of robot systems, which have approximately the same cognitive, communicational, and handling abilities like humans. To achieve this goal, a variety of problems have to be solved by the researchers, for instance in the domain of communication. Here, it is necessary to bridge between different types of communication. On the one side there is the human, who typically prefers a symbolic communication using symbols like gestures or words. On the other side there is the robot system, which typically prefers a subsymbolic communication with subsymbolic information like joint positions, forces, or motor currents.

In general, there is no direct mapping from symbolic to subsymbolic information. Therefore, a transformation is needed, which is able to combine the two types of representations. Therefore, one objective of this work is the analysis of the requirements for such a transformation layer and the proposal of concepts for combining symbolic and subsymbolic representations. Based on this concepts, we describe how to ground symbols to a robot system. At last, we analyze how a robot system can utilize the proposed concepts in order to provide an intuitive, symbolic user interface.

Within this work, we propose the usage of physics, respectively physical effects and quantities, in order to bridge between symbolic and subsymbolic information. These effects and quantities are used for the description of executable actions, respectively for symbolic parameters of executable actions. For the description of executable actions we introduce the concept of *Principal Physical Effects*. This concept describes the relations between the transformation layer and the subsymbolic action representation suitable for a robot system. More specific, this concept is used to describe the relations between a physical process, a physical law, and a subsymbolic action representation based on sensor-based robot motions. To describe the relations between a symbolic representation and the transformation layer, we enhance the concept of Principal Physical Effects to *Verbalized Physical Effects*. This allows a mapping of the symbolic description of a verbalized instruction (e.g. shove, rotate), the symbolic parameter of a verbalized instruction (for instance nominal or prepositional phrases) to a corresponding Principal Physical Effect. To use symbolic parameters combined with an action representation based on Principal Physical Effects, these symbols have to be described in terms of physics. Therefore, we propose the usage of a physical dictionary, which stores the symbol information in context of physical properties.

Besides the grounding of symbol information, we analyze how a robot system can utilize the proposed symbol description in order to provide an intuitive, symbolic user interface. Therefore, we describe additional components, which illustrate the transformation of a symbolic instruction towards a parameterized, subsymbolic robot motion. At last, we evaluate the concepts using a prototype system.

Zusammenfassung

Ein Schwerpunkt der aktuellen Forschung im Bereich der Robotik stellt die Entwicklung von Robotersystemen dar, deren Fähigkeiten in verschiedenen Bereichen stetig komplexer werden und als Fernziel denen des Menschen entsprechen sollen. Betrachtet man beispielsweise die bisherigen Unterschiede im Bereich der Kommunikation, so bevorzugt der Mensch die Kommunikation über Symbole wie Worte oder Gesten, das Robotersystem bevorzugt hingegen die Kommunikation über subsymbolische Information wie Gelenkwinkelstellungen, Kräfte oder Motorströme.

Da im Allgemeinen keine direkte Zuordnung zwischen der symbolischen und subsymbolischen Information möglich ist, muss eine Zwischenrepräsentation gefunden werden, welche die Zusammenhänge der beiden Repräsentationen beschreiben kann. Im Rahmen dieser Arbeit wird zunächst analysiert, in welcher Form eine solche Zwischenrepräsentation umgesetzt werden kann. Darauf aufbauend wird untersucht, in welcher Form sich die zur Kommunikation zur Verfügung stehenden Symbole (Worte) unter Zuhilfenahme der gewählten Zwischenrepräsentation beschreiben und systematisch in einem Robotersystem hinterlegen lassen. Abschließend wird untersucht, in welcher Form unter Nutzung der vorgeschlagenen Konzepte ein intuitiv zu verwendendes Gesamtsystem umgesetzt werden kann.

Im Rahmen dieser Arbeit wird als Zwischenrepräsentation die Verwendung von physikalischen Vorgängen und Eigenschaften vorgeschlagen. Diese werden zur Beschreibung von ausführbaren Aktionen und deren symbolischen Parameter verwendet. Zur Aktionsbeschreibung wird das Konzept der *Prinzipiellen Physikalischen Effekte* vorgestellt. Dieses beschreibt die Zusammenhänge der gewählten Zwischenrepräsentation und der subsymbolischen Aktionsrepräsentation des Robotersystems. Konkret werden die Zusammenhänge zwischen einem physikalischen Vorgang, einer physikalischen Gesetzmäßigkeit und einer sensorbasierten Roboterbewegung definiert. Zur Beschreibung der Zusammenhänge der Zwischenrepräsentation und der symbolischen Repräsentation des Menschen, wird das Konzept der Prinzipiellen Physikalischen Effekte zu *Verbalisierten Physikalischen Effekte* erweitert. Diese beschreiben die Zusammenhänge zwischen der symbolischen Repräsentation in Form einer verbalen Beschreibung (z.B. *schieben*, *drehen*), einem Prinzipiellen Physikalischen Effekt sowie den symbolischen Parameter einer verbal instruierten Anweisung (z.B. Nominal- oder Präpositionalphrasen). Die symbolischen Parameter (bestehend aus Symbolen unterschiedlicher Typen wie z.B. Adjektive oder Nomen) werden ebenfalls anhand physikalischer Eigenschaften repräsentiert. Basierend auf einer Kategorisierung der Symbole wird dargestellt, in welcher Form diese beschrieben und als Teil einer Wissensbasis im Robotersystem hinterlegt werden können.

Neben dem Hinterlegen von Symbolinformation wird weiterhin untersucht, in wie weit basierend auf der vorgeschlagenen Symbolbeschreibung ein intuitiv zu verwendendes Gesamtsystem umgesetzt werden kann. Hierzu werden weitere Komponenten dargestellt, welche die Verarbeitung einer symbolischen Anweisung bis hin zur subsymbolischen und parametrisierten Roboteraktion beschreiben. Abschließend werden die vorgestellten Konzepte im Rahmen eines prototypischen Gesamtsystems umgesetzt und anhand geeigneter Methoden evaluiert.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Aufgabenstellung und Ziele	3
1.3	Abgrenzung	6
1.4	Kapitelübersicht	10
2	Grundlagen und Stand der Forschung	13
2.1	Schnittstellen zur Benutzerinteraktion	14
2.2	Schnittstellen zur Definition von Roboteraktionen	18
2.3	Gesamtsysteme zur intuitiven Roboterkommandierung	23
2.4	Zusammenfassung	34
3	Beschreibung von ausführbaren Roboteraktionen	37
3.1	Ansatz	38
3.2	Prinzipielle Physikalische Effekte	40
3.3	Verbalisierte Physikalische Effekte	57
3.4	Zusammenfassung	65
4	Hinterlegen von Symbolen zur Roboterkommandierung	67
4.1	Kategorien von Symbolen	68
4.2	Physikalisches Wörterbuch	74
4.3	Zusammenfassung	78
5	Parametrisierung von Roboteraktionen unter Nutzung hinterlegter Symbolinformation	79
5.1	Domänenspezifische Sprache	80
5.2	Auflösen von symbolischer Information	88
5.3	Extraktion von subsymbolischer Information	92
5.4	Zusammenfassung	96

6	Evaluation	99
6.1	Prototypische Umsetzung	100
6.2	Flexible Verwendung der Aktionsbeschreibung	104
6.3	Benutzerevaluation	114
6.4	Zusammenfassung	127
7	Schlussbemerkungen	129
7.1	Fazit	129
7.2	Ausblick	134
	Abbildungsverzeichnis	136
	Tabellenverzeichnis	141
	Quellenverzeichnis	143
	Eidesstattliche Versicherung	159

1 Einleitung

In diesem Kapitel wird zunächst motiviert, in wie weit sich Anforderungen an zukünftige Robotersysteme gegenüber bisher eingesetzten Systemen ändern können und welche Problemstellungen dadurch betrachtet und wissenschaftlich untersucht werden müssen (Kapitel 1.1). Darauf aufbauend werden die Ziele und die im Rahmen dieser Arbeit untersuchten wissenschaftlichen Fragestellungen erläutert (Kapitel 1.2). Anschließend werden getroffene Annahmen beschrieben und diskutiert, sowie nicht betrachtete Problemstellungen explizit genannt (Kapitel 1.3). Abschließend wird eine Übersicht der einzelnen Abschnitte dieser Arbeit dargestellt (Kapitel 1.4).

1.1 Motivation

Im Laufe der letzten Jahre lässt sich ein kontinuierlich steigender Markt für Servicerobotersysteme beobachten [IFR16]. Hierbei wird typischerweise zwischen Servicerobotersysteme für den professionellen und privaten Einsatz differenziert. Unter professionellen Einsatz werden beispielsweise Anwendungsdomänen wie das Transportwesen, die Medizin oder das Militär bezeichnet. Dem privaten Einsatz werden beispielsweise Aufgaben aus dem Unterhaltungssektor oder dem Haushalt zugeordnet. Dem Markt der im privaten Umfeld eingesetzten Servicerobotersysteme wird weiteres Wachstumspotential vorhergesagt, so prognostiziert beispielsweise die *International Federation of Robotics (IFR)* in ihrer Jahresstatistik einen Anstieg um 600% für verkaufte Servicerobotersysteme bis zum Jahr 2019 [IFR16].

Betrachtet man die derzeit für den privaten Haushalt konzipierten Modelle, so handelt es sich hierbei im Allgemeinen um Speziallösungen für genau eine Anwendung, beispielsweise in Form von Mäh-, Wisch- oder Staubsaugerroboter. Die verfügbare Produktpalette soll in Zukunft um Systeme erweitert werden, welche eine Vielzahl an Aufgaben flexibel ausführen können. Um solche komplexen Systeme zu realisieren, werden entsprechende Anforderungen an die Fähigkeiten der Robotersysteme gestellt. Da diese von heutigen Systemen im Allgemeinen nicht erfüllt werden, erfordert dies die Weiterentwicklung der Robotersysteme, beziehungsweise der Roboterfähigkeiten. Daher wurde von Seiten der Forschung, der Industrie und der Regierung entsprechender Forschungsbedarf erkannt und im Rahmen der strategischen Richtlinie *Robotics 2020* formuliert [SPARC15]. In dieser Richtlinie werden insbesondere die Roboterfähigkeiten klassifiziert und jeweilige Evolutions- beziehungsweise Komplexitätsstufen für eine Roboterfähigkeit genannt. Insgesamt werden neun Kategorien von Roboterfähigkeiten beschrieben, welche in Abbildung 1.1 dargestellt sind.

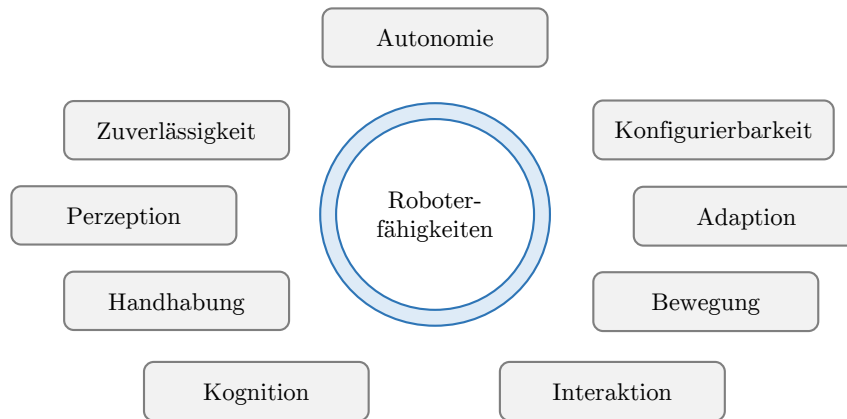


Abbildung 1.1: Übersicht der Fähigkeiten eines Robotersystems, welche zur Erweiterung der Einsatzgebiete für zukünftige Robotersysteme benötigt werden (nach [SPARC15]).

Je nach Einsatzzweck des Robotersystems müssen diverse Fähigkeiten in einer entsprechenden Komplexität vorhanden sein, was im Folgenden anhand eines exemplarischen Robotersystems beschrieben wird, welches für den Einsatz im Rahmen von privaten Haushalten konzipiert werden soll (siehe Abbildung 1.2). Zunächst lässt sich festhalten, dass in diesem Anwendungsbereich das Robotersystem typischerweise nicht durch einen Experten, sondern durch eine Person ohne Expertenwissen im Bereich der Robotik verwendet wird. Daher kann nicht davon ausgegangen werden, dass eine Person ohne Expertenwissen in der Lage ist, alle zur Definition einer ausführbaren Roboteraktion benötigten Parameter festzulegen und explizit zu beschreiben. Vielmehr muss davon ausgegangen werden, dass ein Benutzer lediglich auf die Ausdrucksmöglichkeiten zurückgreifen kann, welche auch im Bereich der zwischenmenschlichen Kommunikation verwendet werden.

Betrachtet man die typische Kommunikation des Menschen, so basiert diese im Normalfall auf *symbolischer* Information, dargestellt durch Symbole wie Worte oder Gesten [White40]. Die typische Kommunikation des Robotersystems basiert hingegen auf *subsymbolischer* Information, dargestellt durch einen abstrakten Befehlssatz mit konkreten Parameterwerten für beispielsweise Arbeitsraumpositionen, Regelsollwerte oder Regelstrategien (z.B. [Finkemeyer04, DeSchutter07]). Soll ein zukünftiges Robotersystem eine intuitive Verwendung durch eine Person ohne Expertenwissen erlauben, so sollte das System eine symbolische Benutzerschnittstelle zur Verfügung stellen, welche an die zwischenmenschliche Kommunikation angelehnt ist, beziehungsweise auf die entsprechenden Kommunikationsmöglichkeiten zurückgreift.

Besitzt das System entsprechende Interaktionsfähigkeiten und stellt eine symbolische und intuitiv zu verwendende Schnittstelle zur Verfügung, erfordert dies weitere Fähigkeiten, beispielsweise aus dem Bereich der Kognition, welche die symbolische Benutzereingabe in eine zur Ausführung geeignete, subsymbolische Repräsentation überführen können. Eine direkte Umwandlung der beiden Formen der Repräsentation ist im Allgemeinen nicht möglich, beziehungsweise nur unter Einschränkungen wie beispielsweise einer im Vorfeld vollständig und präzise bekannten Umgebung umsetzbar (vgl. [Pires05]). In individuellen Umgebungen wie privaten Haushalten ist dies jedoch praktisch nicht umzusetzen.



Abbildung 1.2: Beispiele für Servicerobotersysteme, welche für den Einsatz in Umgebungen wie privaten Haushalten konzipiert werden und über fortschrittliche Fähigkeiten in verschiedenen Bereichen wie beispielsweise der Interaktion, der Kognition oder der Handhabung verfügen müssen (Bildquellen: links [TUM16], rechts [IPA16]).

Daher muss das System zudem in der Lage sein, die zur Ausführung benötigte, subsymbolische Parameterinformation aus der Umgebung zu extrahieren. Dies erfordert zum einen, dass das Robotersystem über zusätzliche Komponenten wie beispielsweise geeignete Sensorik oder Algorithmen verfügen muss. Zum anderen erfordert dies, dass die semantische Information der verwendeten Symbole in einer flexiblen, parametrisierbaren Darstellung im System hinterlegt werden muss. Diese wird durch eine instruierte Anweisung und unter Nutzung der zusätzlichen Komponenten mit konkreten Parameterwerten instantiiert und durch das Robotersystem ausgeführt. Dies erfordert somit zudem eine entsprechende Komplexität für die Roboterfähigkeiten in Bereichen wie beispielsweise der Perzeption oder Handhabung.

Anhand des Beispiels kann gezeigt werden, dass eine Weiterentwicklung und Integration der verschiedenen Roboterfähigkeiten erforderlich ist, um die Einsatzgebiete für Robotersysteme auf die angestrebten Domänen wie beispielsweise private Haushalte zu erweitern und entsprechende Gesamtsysteme zu realisieren. Dies wurde von Seiten der Forschung, Industrie und Regierung erkannt, entsprechender Forschungsbedarf identifiziert und im Rahmen der strategischen Richtlinie *Robotics 2020* beschrieben.

1.2 Aufgabenstellung und Ziele

Wie anhand des Beispiels im Rahmen des vorangegangenen Kapitels beschrieben, erfordert ein für den Einsatz im privaten Umfeld konzipiertes Servicerobotersystem für jeden der Bereiche von Roboterfähigkeiten (siehe Abbildung 1.1) ein Mindestmaß an vorhandener Komplexität. Da die Erweiterung aller Bereiche jedoch typischerweise nicht im Rahmen einer einzelnen Arbeit vorgenommen werden kann, müssen für jede Arbeit die individuell gesetzten Schwerpunkte und die damit gewünschte Weiterentwicklung der Roboterfähigkeiten explizit genannt werden. Zudem ist erforderlich, dass im Rahmen einer Abgrenzung explizit beschrieben wird, welche Komplexitätsebene für jeden Fähigkeitsbereich eines Gesamtsystems angenommen, bzw. verwendet wird (siehe Kapitel 1.3).

Im Rahmen dieser Arbeit soll ein Gesamtsystem umgesetzt werden, welches eine intuitive, symbolische Benutzerschnittstelle zur Verfügung stellt und in der Lage ist, diese in eine zur Ausführung durch das Robotersystem geeignete, subsymbolische Form zu überführen. Da diese Transformation der beiden Repräsentationen im Allgemeinen nicht direkt möglich ist (siehe Kapitel 1.1), wird eine Transformationsschicht benötigt, welche entsprechende Komponenten zur Verfügung stellt.

Daraus resultiert im Allgemeinen eine dreischichtige Systemarchitektur, bestehend aus einer symbolischen Benutzerschicht, einer Transformationsschicht und einer subsymbolischen Kontrollschicht (siehe Abbildung 1.3). Die Benutzerschicht stellt hierbei die eine symbolische Schnittstelle für die Eingaben des Benutzers zur Verfügung. Die Transformationsschicht nimmt eine symbolische Benutzereingabe entgegen und überführt diese anhand von im System vorhandener Komponenten in eine subsymbolische Repräsentation. Diese wird abschließend durch das Robotersystem ausgeführt.

Die Beschreibung der Transformationsschicht inklusive der benötigten Komponenten sowie den entsprechenden Schnittstellen hin zur symbolischen Benutzerschicht sowie der subsymbolischen Kontrollschicht nehmen eine zentrale Stellung im Rahmen dieser Arbeit ein. Die daraus resultierenden Aufgabenpakete und betrachteten wissenschaftlichen Fragestellungen werden im Folgenden aufgezeigt und beschrieben.



Abbildung 1.3: Grafische Darstellung der im Rahmen dieser Arbeit untersuchte Problemstellung in Form einer 3-Schicht-Architektur. Da im Allgemeinen keine direkte Abbildung zwischen symbolischer und subsymbolischer Information möglich ist, wird eine Transformationsschicht zwischen den beiden Repräsentationen benötigt (grün). Die konzeptionelle Beschreibung einer solchen Transformationsschicht nimmt eine zentrale Stellung in dieser Arbeit ein.

Im Rahmen der Transformationsschicht soll die Information der symbolischen Benutzerschnittstelle in eine zur Ausführung geeignete, subsymbolische Repräsentation überführt werden. Im Allgemeinen wird den verwendeten Symbolen eine entsprechende Semantik zugeordnet. Betrachtet man exemplarisch die im Rahmen der natürlichsprachlichen Kommunikation verwendeten Symbole, so existieren hier unterschiedliche Symbole, welche beispielsweise eine auszuführende Aktion (*greifen, bewegen*), ein reales Objekt (*Würfel, Zylinder*) oder eine Objekteigenschaft (*rot, groß*) spezifizieren. Ein Teil der Transformationsschicht muss daher eine Beschreibung der entsprechenden Symbolsemantik zur Verfügung stellen. Eine Herausforderung stellt hierbei die Beschreibung der Semantik von auszuführenden Aktionen dar, da diese typischerweise eine Vielzahl von elementaren Roboterbewegungen und subsymbolischen Parameterwerten erfordern, welche jedoch im Rahmen der symbolischen Beschreibung nicht explizit formuliert werden. Daraus resultiert die wissenschaftliche Fragestellung **F1**, welche wie folgt lautet:

F1 In welcher Form kann die Semantik einer ausführbaren Aktion systematisch beschrieben und in einem Robotersystem hinterlegt werden?

Hierfür muss ein Ansatz gefunden werden, welcher die Semantik einer auszuführenden Aktion in geeigneter Form beschreiben kann. Insbesondere muss die Aktionsbeschreibung in einer parametrisierbaren Form im System hinterlegt werden können. Dies ist notwendig, da neben den Symbolen zur Beschreibung einer auszuführenden Aktion im Rahmen einer symbolischen Anweisung typischerweise weitere Symbole verwendet werden. Betrachtet man beispielsweise die symbolische Anweisung *Berühre den roten Zylinder* so werden hier neben dem Symbol zur Beschreibung der auszuführenden Aktion (*Berühre*) weitere Symbole genutzt, welche die auszuführende Aktion in Form von symbolischen Ergänzungen parametrisieren (*den roten Zylinder*). Daher müssen auch diese Symbole in einer geeigneten Form repräsentiert werden, was im Rahmen der wissenschaftlichen Fragestellung **F2** formuliert wird:

F2 In welcher Form kann die Semantik der zur Parametrisierung einer ausführbaren Aktion verwendeten Symbole systematisch beschrieben und in einem Robotersystem hinterlegt werden?

Im nächsten Schritt können die vorgeschlagenen Konzepte zur Beschreibung der Symbolsemantik integriert werden um gemeinsam innerhalb eines Gesamtsystems verwendet zu werden. Da im Rahmen der Benutzerschnittstelle ausschließlich symbolische Information durch den Benutzer übermittelt wird, das Robotersystem jedoch subsymbolische Parameter zur Ausführung der instruierten Anweisung benötigt, muss das System in der Lage sein, diese über geeignete Komponenten aus der Umgebung zu extrahieren. Daher ist zu analysieren, welche zusätzlichen Komponenten zur Ausführung von symbolisch instruierten Anweisungen im Robotersystem vorhanden sein müssen und in welcher Form diese zusammen mit den Konzepten zur Beschreibung der Symbolsemantik zur Parametrisierung von ausführbaren Roboteraktionen genutzt werden können. Daraus wird die wissenschaftliche Fragestellung **F3** abgeleitet, welche wie folgt lautet:

F3 In welcher Form können die vorgeschlagenen Konzepte zur Beschreibung der Symbolsemantik gemeinsam verwendet werden um eine subsymbolische Parametrierung der auszuführenden Roboteraktionen zu ermöglichen?

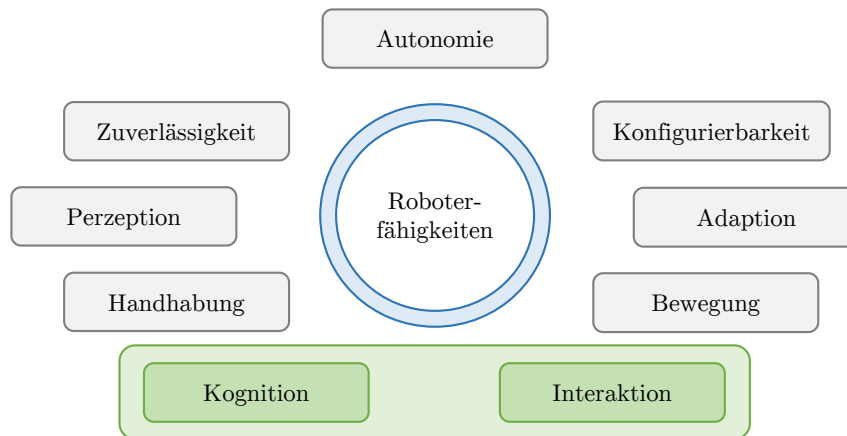


Abbildung 1.4: Übersicht der Fähigkeiten eines Robotersystems, deren Komplexität im Rahmen dieser Arbeit durch eigene wissenschaftliche Beiträge weiterentwickelt werden (grün).

Die im Rahmen der Fragestellungen vorgeschlagenen Lösungsansätze sollen abschließend in Form eines Gesamtsystems umgesetzt werden. Dieses wird zur Evaluation der vorgeschlagenen Konzepte genutzt, wodurch beispielsweise die intuitive Verwendung der vom System zur Verfügung gestellten symbolischen Benutzerschnittstelle evaluiert werden soll. Daraus lässt sich die wissenschaftliche Fragestellung **F4** extrahieren, welche wie folgt lautet:

F4 In wie weit kann basierend auf den vorgeschlagenen Konzepten ein intuitiv zu verwendendes Gesamtsystem umgesetzt werden?

Somit werden insgesamt vier wissenschaftliche Fragestellungen formuliert, welche im Rahmen dieser Arbeit untersucht werden. Die auf Basis der wissenschaftlichen Fragestellungen durchzuführenden Arbeiten haben daher eine Erweiterung der Roboterfähigkeiten in den Bereichen der Kognition und Interaktion zum Ziel (siehe Abbildung 1.4). Hierbei werden die Problemstellungen bezüglich der Abbildung zwischen symbolischer und subsymbolischer Information dem Bereich der Kognition und die Problemstellungen bezüglich der intuitiven Verwendung dem Bereich der Interaktion zugeordnet.

1.3 Abgrenzung

Im Rahmen dieser Arbeit sollen die Fähigkeiten eines Robotersystems in den Bereichen Interaktion und Kognition weiterentwickelt werden (siehe Abbildung 1.4). Eine isolierte Weiterentwicklung einzelner Fähigkeiten ist jedoch typischerweise nicht möglich, da im Allgemeinen eine Vielzahl an Abhängigkeiten zu angrenzenden Fähigkeitsbereichen vorhanden sind und daher für diese Fähigkeiten ein Mindestmaß an Komplexität erforderlich ist. Die angrenzenden Fähigkeitsbereiche der Interaktion und Kognition sind im Wesentlichen die Perzeption, Handhabung, Bewegung und Autonomie. Diese besitzen wiederum Abhängigkeiten zu weiteren Bereichen wie der Zuverlässigkeit oder Adaption (vgl. [SPARC15]). Im Folgenden wird beschrieben, welche Komplexität in den entsprechenden Bereichen erreicht, beziehungsweise angenommen wird.

Um zunächst die im Rahmen dieser Arbeit umzusetzende Interaktion genauer zu spezifizieren, bzw. abgrenzen zu können, wird diese anhand einer geeigneten Menge an Kriterien beschrieben. Es existieren eine Reihe an Arbeiten, welche verschiedene Merkmale zur Beschreibung der Mensch-Roboter-Interaktion vorschlagen [Yanco04, Goodrich07, SPARC15, Onnasch16]. Im Rahmen dieser Arbeit wird die Beschreibung nach [Onnasch16] verwendet und falls nötig an den entsprechenden Stellen ergänzt, beziehungsweise genauer spezifiziert.

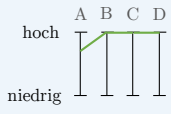
<u>Interaktionsrolle des Menschen</u>	<u>Einsatzgebiet des Roboters</u>	<u>Aufgabe des Roboters</u>	<u>Kommunikationskanal</u>	<u>Zusammensetzung</u>
Nicht-Beteiligter Supervisor Operateur Kooperateur Kollaborateur	Industrie Kom. Service Pers. Service	Informationsaustausch Präzision Entlastung Transport Manipulation	Akustisch Elektronisch Haptisch Mechanisch Taktile Visuell	$N_M = N_R$ $N_M > N_R$ $N_M < N_R$
<u>Interaktionsform</u>	<u>Autonomiegrad des Roboters</u>	<u>Morphologie des Roboters</u>	<u>Räumliche Nähe</u>	<u>Zeitliche Nähe</u>
Ko-Existenz Kooperation Kollaboration	 <p>hoch niedrig</p> <p>A B C D</p> <p>A: Aufnahme B: Verarbeitung C: Entscheidung D: Ausführung</p>	Funktional Humanoid Zoomorph	Ferngesteuert Vermeidend Vorbeigehend Führend Annähernd Berührend	Asynchron Synchron

Abbildung 1.5: Grafische Darstellung der im Rahmen dieser Arbeit umzusetzenden Mensch-Roboter-Interaktion. Die Merkmale zur Beschreibung einer Mensch-Roboter-Interaktion sind aus [Onnasch16] entnommen. Die verwendeten Merkmale werden kategorisiert in Merkmale zur Interaktions- (erste Spalte), Team- (zweite und dritte Spalte) und Roboterklassifikation (vierte und fünfte Spalte).

Die im Rahmen dieser Arbeit umzusetzende Mensch-Roboter-Interaktion ist in Abbildung 1.5 dargestellt und wird im Folgenden erläutert. Wie bereits im letzten Kapitel aufgezeigt, beschreibt ein Ziel dieser Arbeit die im System vorhandene Funktionalität in Form einer intuitiv zu verwendenden, symbolischen Benutzerschnittstelle zur Verfügung zu stellen. Hierbei soll der Benutzer dem System einzelne Anweisungen instruieren, welche durch das Robotersystem ausgeführt werden. Daher wird die Rolle des Menschen bei der Interaktion als Operateur festgelegt. Da durch die gegebenen Instruktionen ein gemeinsames Ziel verfolgt wird, ist die Form der Interaktion als Kooperation festzulegen. An dieser Stelle wird explizit darauf hingewiesen, dass der Mensch ausschließlich als Operateur agiert, das heißt nur das Robotersystem ist bei der Umsetzung des gemeinsamen Ziels ausführend tätig.

Zur Klassifikation des Robotersystems werden zunächst die potentiellen Einsatzgebiete festgelegt. Wie in Kapitel 1.1 genannt, wird eine Erweiterung der Anwendungsdomänen zukünftiger Robotersysteme auf beispielsweise private Haushalte angestrebt. Daher ist als Einsatzgebiet der Bereich der persönlichen Servicerobotik, als Aufgaben der Bereich der Manipulation festzulegen (vgl. Abbildung 1.2).

In diesen Anwendungsbereichen sind verschiedene Typen von Robotersystemen denkbar, in dieser Arbeit wird sich auf ein stationäres Robotersystem beschränkt, welches über einen Roboterarm sowie der zur Ausführung von symbolisch instruierten Anweisungen benötigten Sensorik verfügt. Somit ist die verwendete Morphologie des Roboters als funktional festzulegen.

Durch die zu entwickelnden Konzepte soll eine weitestgehend autonome Ausführung der symbolisch instruierten Anweisungen ermöglicht werden. Daher wird der Autonomiegrad für die Informationsverarbeitung, die Entscheidungsfindung und die Handlungsausführung als hoch festgelegt. Da der Benutzer die auszuführende Instruktion dem System übermitteln muss, die zur Ausführung benötigten Parameter jedoch autonom aus der Umgebung extrahiert werden sollen, wird im Bereich der Informationsaufnahme keine volle Autonomie erreicht.

Zum Abschluss werden die Merkmale im Bereich der Teamklassifikation beschrieben. Im Rahmen dieser Arbeit wird angenommen, dass es zur Laufzeit genau einen Benutzer und genau ein zu instruierendes Robotersystem gibt. Da der Mensch ausschließlich Anweisungen instruiert, welche daraufhin durch das Robotersystem ausgeführt werden, wird die zeitliche Nähe als asynchron festgelegt. An dieser Stelle sei explizit ausgeschlossen, dass während der Aktionsausführung weitere Anweisungen instruiert werden. Dies erfordert erweiterte Komponenten zur Koordination der Ausführung, deren Umsetzung nicht im Fokus dieser Arbeit stehen.

Die räumliche Nähe wird als vermeidend festgelegt, da sich Mensch und Robotersystem prinzipiell in einem gemeinsamen Arbeitsraum aufhalten können. Jedoch ist im Rahmen dieser Arbeit allein der Mensch für die Kollisionsvermeidung zuständig. Zudem ist der Benutzer des Systems verantwortlich, dass die instruierten Anweisungen durch das Robotersystem ausführbar sind.

Zur Kommunikation zwischen Mensch und Roboter wird der elektronische Kanal verwendet. Dieser entspricht gemäß [Onnasch16] der Eingabe durch Steuerelemente und im Rahmen dieser Arbeit einer Eingabe in textueller Form via Tastatur. Zur Rückmeldung des Robotersystems an den Mensch wird der mechanische und visuelle Kommunikationskanal verwendet. In diesem Kontext stellt der mechanische Kommunikationskanal die Ausführung der instruierten Anweisung und der visuelle Kommunikationskanal eine Rückmeldung in Form einer Bildschirmanzeige des Systemzustands dar.

<u>Interaktionsschnittstelle</u> Subsymbolisch Symbolisch	<u>Interpretation</u> Vordefiniert Benutzereingaben Objektparameter Aktionsparameter Vorausschauend	<u>Urteilsvermögen</u> Vordefiniert Konfliktorientiert Sicherheitsorientiert Aufgabenorientiert
<u>Ausführung</u> Vordefiniert Parametrisierbar Wissensbasiert Vorausschauend	<u>Hinterlegte Information</u> Vollständig Unvollständig	<u>Lernvermögen</u> Vordefiniert Aufgabenorientiert Sicherheitsorientiert Benutzerorientiert

Abbildung 1.6: Grafische Darstellung der im Rahmen dieser Arbeit umzusetzenden kognitiven Fähigkeiten. Die Merkmale zur Beschreibung sind angelehnt an [SPARC15] und werden unterteilt in Merkmale zur Ein- und Ausgabeklassifikation (erste Spalte) sowie der Systemklassifikation (zweite und dritte Spalte).

Neben den Fähigkeiten im Bereich Interaktion sollen auch die zu erreichenden Fähigkeiten im Bereich der Kognition spezifiziert, bzw. abgegrenzt werden (siehe Abbildung 1.6). Zur Beschreibung werden Kriterien verwendet, welche an die Merkmale aus [SPARC15] angelehnt sind.

Im Rahmen dieser Arbeit wird zur Interaktion eine symbolische Schnittstelle verwendet, welche die durch den Benutzer instruierte und durch das Robotersystem auszuführende Aktion beschreibt. Die genaue Ausprägung der symbolischen Schnittstelle ist noch festzulegen.

Dem System soll neben der instruierten Anweisung eine semantische Beschreibung der verwendeten Symbole zur Verfügung stehen. Da das System jedoch die zur Ausführung benötigte subsymbolische Information zunächst aus der Umgebung extrahieren muss, ist der Bereich der hinterlegten Information als unvollständig zu klassifizieren. Im Bereich der Interpretation ist es nötig, entsprechende Aktionsparameter zu interpretieren.

Die aufgrund der perzeptiven Fähigkeiten aus der Umgebung extrahierte subsymbolische Information wird zur Parametrisierung der auszuführenden Aktion verwendet. Daher ist im Rahmen dieser Arbeit eine wissensbasierte Ausführung umzusetzen. Die Bereiche Lern- und Urteilsvermögen werden im Rahmen dieser Arbeit nicht betrachtet.

Abschließend werden die angenommenen Komplexitätsebenen für die im Rahmen dieser Arbeit nicht weiter zu entwickelnden Fähigkeitsbereiche beschrieben. Diese werden ausschließlich zur Vollständigkeit erläutert, eine detaillierte Darstellung der verschiedenen Fähigkeitsbereiche ist in [SPARC15] zu finden.

Im Bereich der Konfigurierbarkeit wird eine statische Konfiguration des Robotersystems angenommen, das heißt die zur Verfügung stehenden Hard- und Softwarekomponenten des Robotersystems sind zur Laufzeit bekannt. Es ist jedoch ein Ziel, die zu entwickelnden Konzepte unabhängig von konkreten Systemkonfigurationen zu gestalten.

Im Bereich der Zuverlässigkeit wird ausschließlich ein sicherer Abbruch der aktuellen Aktion im Falle einer Fehlersituation gefordert. Daraus folgt, dass der Benutzer des Prototyps eine prinzipielle Ausführung der auszuführenden Aktion gewährleisten muss.

Im Bereich der Handhabungs- und Bewegungsfähigkeiten soll das System in der Lage sein, sensorbasierte Bewegungen auszuführen sowie Objekte zu manipulieren. Dies stellt entsprechende Anforderungen an das verwendete Robotersystem wie beispielsweise einer Greifvorrichtung oder geeigneter Sensorik zur Ausführung der sensorbasierten Bewegungen.

Im Bereich der Perzeption sollen dem System geeignete Sensorik sowie geeignete Softwarekomponenten zur Verfügung stehen, welche es erlauben, die zur Ausführung benötigte, subsymbolische Information aus der Umgebung zu extrahieren.

Die Bereiche Autonomie und Adaption sind bereits implizit durch die umzusetzende Kognition und Interaktion beschrieben. Im Bereich der Autonomie muss das System in der Lage sein, die benötigte subsymbolische Information aus der Umgebung zu extrahieren. Im Bereich der Adaption müssen die extrahierten Größen bei der Anweisungsausführung berücksichtigt werden.

1.4 Kapitelübersicht

Nachdem die im Rahmen dieser Arbeit wissenschaftlich untersuchten Fragestellungen motiviert und beschrieben wurden, wird abschließend die Struktur und Inhalte der folgenden Kapitel dargestellt (siehe Abbildung 1.7).

Wie in Kapitel 1.2 beschrieben, nimmt die Beschreibung der Zusammenhänge von symbolischer und subsymbolischer Information und die dadurch benötigten Mechanismen zur Umwandlung der Repräsentationen einen wesentlichen Aspekt dieser Arbeit ein. Die wissenschaftlichen Fragestellungen umfassen dabei die systematische Beschreibung von Roboteraktionen, das Hinterlegen von Symbolsemantik, die Parametrisierung von Roboteraktionen sowie die Umsetzung einer intuitiv zu verwendenden Benutzerschnittstelle.

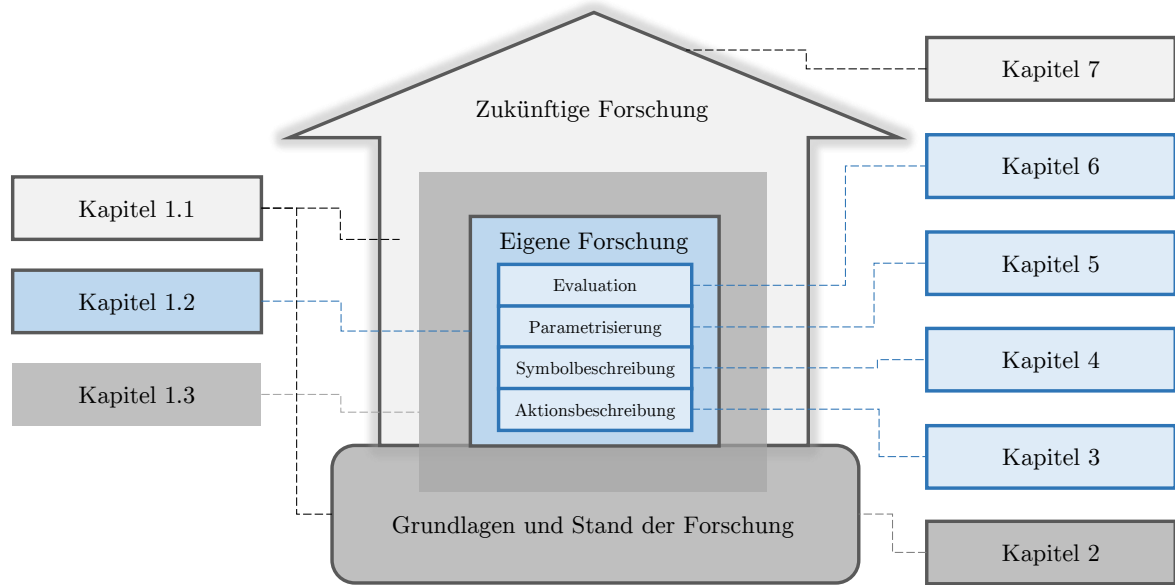


Abbildung 1.7: Schematische Zuordnung der einzelnen Kapitel dieser Arbeit.

Im folgenden Kapitel 2 werden daher zunächst die benötigten Grundlagen sowie der Stand der Forschung in den entsprechenden Bereichen dargestellt. Dies umfasst zum einen eine Übersicht verschiedener Schnittstellen zur Benutzerinteraktion (Kapitel 2.1). Zum anderen umfasst dies eine Darstellung von Schnittstellen, welche typischerweise innerhalb eines Robotersystems zur Definition von ausführbaren Roboterbewegungen verwendet werden (Kapitel 2.2). Des Weiteren werden Gesamtsysteme zur intuitiven Roboterkommandierung aufgezeigt und anhand geeigneter Kriterien bewertet und analysiert (Kapitel 2.3). Abschließend werden die extrahierten Schlussfolgerungen beschrieben und diskutiert (Kapitel 2.4).

In Kapitel 3 wird zunächst dargestellt, in welcher Form die ausführbaren Roboteraktionen im Rahmen dieser Arbeit definiert werden. Dies umfasst zunächst eine Beschreibung des Grundansatzes (Kapitel 3.1). Darauf aufbauend wird anschließend das Konzept der *Prinzipiellen Physikalischen Effekte* eingeführt und beschrieben (Kapitel 3.2). Dieses stellt die Zusammenhänge zwischen der subsymbolischen Aktionsbeschreibung und der gewählten Zwischenrepräsentation dar. Um die symbolische Repräsentation mit der gewählten Zwischenrepräsentation zu verbinden, wird das Konzept der Prinzipiellen Physikalischen Effekte erweitert zu *Verbalisierten Physikalischen Effekten*, welches im darauf folgenden Kapitel beschrieben wird (Kapitel 3.3). Abschließend folgt eine Zusammenfassung und Diskussion der vorgeschlagenen Konzepte (Kapitel 3.4).

Im Rahmen von Kapitel 4 wird beschrieben, wie zusätzliche Symbole zur Roboterkommandierung im Robotersystem hinterlegt werden können. Diese Symbole werden typischerweise zur Parametrisierung einer verbal instruierten Anweisung verwendet und spezifizieren hierbei beispielsweise die im Rahmen der Anweisung benötigten Objekte. Zunächst werden die zur Verfügung stehenden Symbole kategorisiert und auf deren Funktion im Rahmen einer verbalisierten Anweisung untersucht (Kapitel 4.1).

Aufbauend auf diesen Erkenntnissen wird dargestellt, wie die Semantik der Symbole beschrieben und im Robotersystem hinterlegt werden kann (Kapitel 4.2). Analog zum vorangegangenen Kapitel erfolgt auch hier abschließend eine Zusammenfassung und Diskussion der vorgeschlagenen Konzepte (Kapitel 4.3).

Im Rahmen von Kapitel 5 wird beschrieben, wie die vorgeschlagenen Konzepte genutzt werden können, um eine Parametrisierung und Ausführung von Roboteraktionen anhand der hinterlegten Symbolinformation zu ermöglichen. Hierzu wird zunächst eine symbolische Schnittstelle in Form einer domänenspezifischen Sprache eingeführt (Kapitel 5.1). Anhand dieser wird gezeigt, wie das Robotersystem die symbolische Information einer instruierten Anweisung auflösen und in die gewählte Zwischenrepräsentation überführen kann (Kapitel 5.2). Als nächster Schritt hin zur vollständigen Parametrisierung einer Roboteraktion wird beschrieben, in wie weit das Robotersystem die zur Ausführung benötigte subsymbolische Information unter Nutzung von der hinterlegten Symbolinformation sowie zusätzlichen Komponenten aus der Umgebung extrahieren kann (Kapitel 5.3). Analog erfolgt auch hier abschließend eine Zusammenfassung und Diskussion (Kapitel 5.4).

Im Rahmen von Kapitel 6 werden die vorgeschlagenen Konzepte anhand einer prototypischen Umsetzung evaluiert. Hierzu wird zunächst der Aufbau des Systems sowie die verwendeten Hard- und Softwarekomponenten beschrieben (Kapitel 6.1). Anschließend wird die flexible Verwendung der entwickelten Aktionsbeschreibung auf verschiedenen Systemkonfigurationen analysiert (Kapitel 6.2). Im Rahmen einer Benutzerevaluation wird untersucht, in wie weit das vorgeschlagene System als intuitiv durch den Benutzer wahrgenommen wird (Kapitel 6.3). Die gewonnenen Erkenntnisse werden abschließend zusammengefasst und diskutiert (Kapitel 6.4).

Zum Abschluss der Arbeit werden in Kapitel 7 die vorgeschlagenen, umgesetzten und evaluierten Konzepte zur Verbindung von symbolischer und subsymbolischer Information zusammengefasst (Kapitel 7.1) sowie mögliche Erweiterungen beschrieben (Kapitel 7.2).

2 Grundlagen und Stand der Forschung

Im Rahmen dieses Kapitels werden die benötigten Grundlagen bzw. die verwandten Arbeiten erläutert. Wie bereits in Kapitel 1.2 beschrieben, beschäftigt sich ein Teil dieser Arbeit mit den Zusammenhängen von symbolischer und subsymbolischer Information. Die Verwendung von symbolischer Information im Rahmen der Benutzerschnittstelle soll eine intuitive Nutzung des Robotersystems ermöglichen, insbesondere für Personen ohne Expertenwissen im Bereich der Robotik. Die subsymbolische Schnittstelle hin zum Robotersystem wird zur Definition und Ausführung von sensorbasierten Bewegungen benötigt.

Im Folgenden werden zunächst die beiden an die Transformationsschicht angrenzenden Bereiche der symbolischen Benutzerschicht (Kapitel 2.1) und der subsymbolischen Kontrollschicht (Kapitel 2.2) erläutert. Im Bereich der symbolischen Schnittstellen zur Benutzerinteraktion wird ein Überblick über die im Rahmen der zwischenmenschlichen Kommunikation verwendeten Eingabemodalitäten präsentiert. Anschließend werden die bereits technisch umgesetzten Lösungen beschrieben und bewertet. Im Bereich der subsymbolischen Schnittstellen zur Definition von ausführbaren Roboteraktionen werden zunächst die generellen Möglichkeiten zur Aktionsbeschreibung dargestellt, anschließend wird die im Rahmen dieser Arbeit verwendete Schnittstelle im Detail beschrieben.

Im Anschluss werden die bereits vorhandenen Gesamtsysteme zur intuitiven Roboterkommandierung dargestellt und anhand geeigneter Kriterien kategorisiert und bewertet (Kapitel 2.3). Zunächst wird eine Übersicht der vorhandenen Systeme auf Basis der zur Verfügung stehenden Benutzer- und Kontrollschicht beschrieben. Dies erlaubt eine grobe Kategorisierung des Stands der Forschung sowie eine Zuordnung des im Rahmen dieser Arbeit umgesetzten Systems. An dieser Stelle wird die umgesetzte Transformationsschicht zunächst als Blackbox betrachtet und es werden die in Abhängigkeit der Benutzer- und Kontrollschicht prinzipiellen Komponenten der Systeme beschrieben. Anschließend werden die Möglichkeiten zur Umsetzung der Transformationsschicht im Detail betrachtet. Auch an dieser Stelle werden zunächst die prinzipiellen Komponenten der Transformationsschicht dargestellt. Daraufhin werden die einzelnen Arbeiten kurz beschrieben und anhand einer Menge an geeigneten Merkmalen kategorisiert und bewertet.

Abschließend werden die gewonnenen Erkenntnisse zusammengefasst und die daraus extrahierten Schlussfolgerungen beschrieben und diskutiert (Kapitel 2.4).

2.1 Schnittstellen zur Benutzerinteraktion

In diesem Kapitel werden zunächst die Möglichkeiten zur Umsetzung der symbolischen Benutzerschicht beschrieben (siehe Abbildung 2.1). Als Basis dienen hierbei die im Rahmen der zwischenmenschlichen Kommunikation eingesetzten Wahrnehmungssysteme, welche im Folgenden dargestellt werden. Anschließend werden die bereits in Systemen zur Mensch-Roboter-Interaktion verwendeten Methoden aufgezeigt.



Abbildung 2.1: Inhaltliche Zuordnung des Kapitels im Gesamtsystem.

Versteht man den Menschen als technisches System, so stehen ihm eine Reihe an Sensoren zur Verfügung, um Informationen aus der Umgebung wahrzunehmen. Diese entsprechen den Sinnesorganen, welche es erlauben, physikalische Reize zu registrieren und zu verarbeiten. Daher werden diese auch als sensorische Modalitäten [Luczak93] oder Sinnesmodalitäten [Roetting05] bezeichnet. Eine Übersicht der sensorischen Modalitäten des Menschen nach [Luczak93] beziehungsweise [Roetting05] ist in Tabelle 2.1 dargestellt.

Betrachtet man die im Rahmen einer Mensch-Mensch-Interaktion verwendeten Wahrnehmungssysteme, so sind dies typischerweise das auditive, taktile und visuelle Wahrnehmungssystem. Diese entsprechen auch den primär eingesetzten Kommunikationskanälen im Bereich der Mensch-Roboter-Interaktion [Goodrich07]. Auf Basis der genannten Wahrnehmungsmechanismen gibt es eine Vielzahl von Systemen, welche eine Kommunikation zwischen Benutzer und Robotersystem unter Nutzung des entsprechenden Kommunikationskanals erlauben.

Das visuelle Wahrnehmungssystem des Robotersystems wird typischerweise dazu verwendet, Benutzereingaben in Form von Gesten entgegenzunehmen. Es existieren bereits Lösungen zur Erkennung eines breiten Spektrum an Gesten, darunter beispielsweise Hand- bzw. Armgesten [Suarez12,Rautaray15], Gesichtsgesten [Hjelmas01,Zhao03] oder Ganzkörpergesten [Moeslund01,Gavrila99].

Wahrnehmungssystem	Reiz	Organ	Empfindung
Auditiv	Periodische Luftdruckschwankungen	Innenohr	Tonhöhe, Lautstärke
Gustatorisch	Moleküle in Flüssigkeiten	Zunge	Geschmack
Kinästhetisch	Dehnung der Muskeln und Bänder	Muskelspindeln	Körperstellung
Olfaktorisch	Moleküle in Gasen	Nase	Geruch
Schmerz	Verletzung oder Belastung	Nervenenden	Schmerz
Taktil	Verformungen der Haut	Haut	Druck, Berührung
Thermisch	Temperatur	Haut	Wärme
Vestibulär	Flüssigkeitsverschiebung und Statolithen	Mittelohr	Linear- und Winkelbeschleunigung
Visuell	Elektromagnetische Strahlung	Auge	Farbe, Helligkeit

Tabelle 2.1: Sensorische Modalitäten des Menschen gemäß [Luczak93] bzw. [Roetting05]. Die blau hervorgehobenen Einträge stellen die Modalitäten dar, welche typischerweise zur Benutzerinteraktion im Rahmen eines technischen Systems verwendet werden.

Das taktile Wahrnehmungssystem wird zur Umsetzung von Schnittstellen benötigt, welche eine direkte physikalische Interaktion mit dem System erfordern. Das Spektrum der umgesetzten Schnittstellen reicht von der Verwendung von einfachen Steuergeräten wie einer Tastatur bis hin zu technisch aufwändigen Lösungen wie beispielsweise druck-sensitiven Displays oder Sensorhäuten. Eine Übersicht vorhandener Systeme, welche eine Nutzerinteraktion unter Verwendung des taktilen Wahrnehmungssystems des Roboters umsetzen, ist beispielsweise [Benali04] oder [Argall10] zu entnehmen.

Das auditive Wahrnehmungssystem wird im Allgemeinen zur Erkennung von Sprache verwendet. Dabei ist es typischerweise das Ziel, das vom Benutzer abgesetzte Sprachsignal zu erkennen und in eine textuelle Darstellung zu überführen, welche daraufhin zur Weiterverarbeitung genutzt wird. Im Bereich der *Spracherkennung* wird beispielsweise die Verarbeitung des Sprachsignals sowie das Erkennen einzelner Laute bzw. darauf aufbauend die Erkennung von Worten vorgenommen. Die Weiterverarbeitung der textuellen Repräsentation wird im Allgemeinen als *Sprachverarbeitung* bezeichnet und umfasst beispielsweise die Gliederung der Worte in Sätze und Satzbestandteile oder die Bestimmung der Wortart eines Symbols (vgl. [Jurafsky09]). Wie auch in den anderen Bereichen gibt es bereits eine Reihe an Systemen, welche eine Nutzung von Sprache zur Benutzerinteraktion erlauben. Diese sind auch außerhalb der Robotik vorzufinden, beispielsweise in Form von Smartphones [Apple16, Microsoft16]. Im Bereich der Robotik werden diese Schnittstellen unter anderem zur Steuerung von stationären [Laengle95, Knoll97] oder mobilen Systemen [Lauria02, Kollar10] eingesetzt.

Wie bereits im Kapitel 1.3 beschrieben, soll der Benutzer im Rahmen dieser Arbeit umgesetzten Mensch-Roboter-Interaktion als Operator dienen, das heißt er soll dem System die auszuführenden Aktionen in Form einzelner Instruktionen übermitteln. Die Instruktion der Anweisungen kann beispielsweise unter Nutzung des auditiven oder taktilen bzw. elektronischen (gemäß [Onnasch16]) Wahrnehmungssystems des Roboters durchgeführt werden. Die Nutzung des auditiven Wahrnehmungssystems entspricht einer Eingabe in Form von Sprachanweisungen und die Nutzung des taktilen, bzw. elektronischen Wahrnehmungssystems entspricht einer textuellen Eingabe durch ein Steuergerät wie beispielsweise einer Tastatur. Da ein Sprachsignal zur Weiterverarbeitung im Allgemeinen in eine textuelle Darstellung überführt wird und die Eingabe via Tastatur ebenfalls in einer textuellen Darstellung vorhanden ist, wird im Rahmen dieser Arbeit eine solche textuelle Repräsentation als symbolische Schnittstelle zwischen der Benutzerschicht und Transformationsschicht verwendet. Da eine textuelle Darstellung jedoch in verschiedenen Ausprägungen vorliegen kann, werden diese im Folgenden beschrieben und die Unterschiede anhand eines Beispiels erläutert.

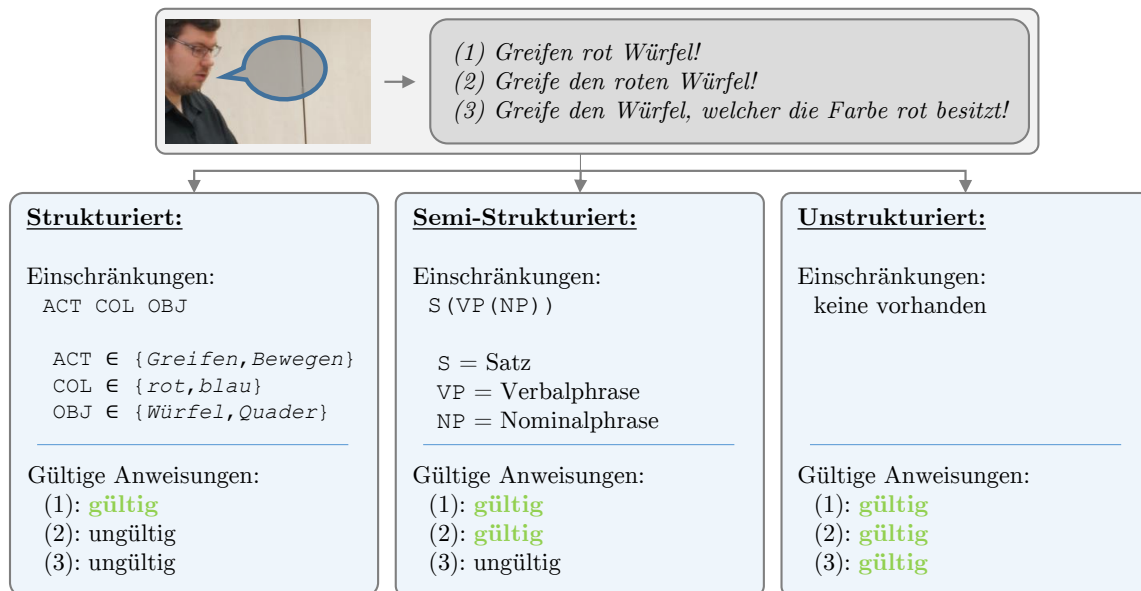


Abbildung 2.2: Beispiel zur Verdeutlichung der Unterschiede zwischen den verschiedenen Formen (strukturiert, semi-strukturiert und unstrukturiert) einer textuellen Repräsentation.

Im Allgemeinen kann die Art einer textuellen Repräsentation als *strukturiert*, *semi-strukturiert* oder *unstrukturiert* beschrieben werden [Aggarwal12]. Bei einer strukturierten Beschreibung sind ausschließlich Anweisungen zulässig, welche beispielsweise einer vorgegebenen Syntax oder einem bekannten Schema genügen. Im Gegensatz dazu werden bei einer unstrukturierten Beschreibung keine Einschränkungen an die Anweisungen beschrieben. Eine semi-strukturierte Anweisung ist konzeptuell zwischen den beiden zuvor genannten Varianten angesiedelt und beschreibt Daten, welche keiner festen Syntax gehorchen, jedoch eine prinzipielle Struktur besitzen. Zur Verdeutlichung der Unterschiede zwischen den drei Arten von textuellen Repräsentationen ist ein Beispiel in Abbildung 2.2 dargestellt.

Ist das System ausschließlich in der Lage strukturierte Eingaben zu verarbeiten, so können beispielsweise die in Abbildung 2.2 (links) dargestellten Einschränkungen im System hinterlegt sein. Hier wird beschrieben, dass eine Anweisung ausschließlich gültig ist, wenn sie exakt der definierten Reihenfolge mit exakt festgelegten Parameterwerten genügt. Daraus folgt, dass bei den genannten Einschränkungen lediglich Anweisung (1) als gültig erkannt wird. Soll das System in der Lage sein, semi-strukturierte Eingaben zu verarbeiten, können die Einschränkungen weniger strikt beschrieben werden. Beispielsweise werden keine Einschränkungen in Bezug auf einzelne Parameter, sondern auf die allgemeine Struktur der Eingabe formuliert (siehe Abbildung 2.2 (mitte)). Im Beispiel werden alle Anweisungen als gültig erkannt, welche sich aus einer Verbalphrase mit darin eingebetteter Nominalphrase zusammensetzen. Daher sind die Anweisungen (1) und (2) als gültig zu klassifizieren. Soll das System in der Lage sein, unstrukturierte Daten verarbeiten zu können, sind im Allgemeinen keine Einschränkungen an die Anweisungen vorhanden und somit werden alle Anweisungen als gültig klassifiziert (siehe Abbildung 2.2 (rechts)). An dieser Stelle sei jedoch darauf hingewiesen, dass dies auch für Anweisungen zutreffen würde, welche keinen ersichtlichen Sinn ergeben wie *"Würfel greife 123 den rot!"*.

Bewertet man die verschiedenen Arten einer textuellen Repräsentation, so besteht der Vorteil bei der Verwendung von strukturierten Eingaben darin, dass eine einfache Zuordnung zwischen instruierbarer und relevanter Information möglich ist. Ein Nachteil stellt jedoch die fehlende Flexibilität für den Benutzer dar, da ausschließlich Anweisungen zulässig sind, welcher der definierten Syntax genügen. Die volle Flexibilität steht zur Verfügung, wenn das System in der Lage ist, unstrukturierte Eingaben zu verarbeiten. Jedoch bedingt dies eine höhere Komplexität im Bereich der verfügbaren Verfahren zur Eingabeverarbeitung. Daher stellt die Verwendung von semi-strukturierten Daten typischerweise einen Kompromiss zwischen Flexibilität bei der Eingabe und der Komplexität der zur Verarbeitung benötigten Komponenten dar.

Abschließend lässt sich festhalten, dass zur Umsetzung einer symbolischen Benutzerschnittstelle in erster Linie das auditive, taktile oder visuelle Wahrnehmungssystem des Robotersystems verwendet wird. Betrachtet man die Eingaben in Form von Sprache, Tastatureingaben oder Gesten, so werden die Eingangssignale typischerweise in eine interne, textuelle Repräsentation überführt. Während diese bei der Verwendung von Gesten meist in einer strukturierten Form beschrieben wird, kann diese bei Verwendung anderer Eingabeformen als semi-strukturierte oder unstrukturierte Anweisung vorliegen. Je nach verfügbaren Verfahren zur Eingabeverarbeitung sind die Systeme in der Lage, strukturierte, semi-strukturierte oder unstrukturierte Anweisungen zu verarbeiten. Im Rahmen dieser Arbeit wird eine semi-strukturierte Darstellung der auszuführenden Instruktionen verwendet, welche einen Kompromiss zwischen der Flexibilität bei der Eingabe und der Komplexität der zur Verarbeitung benötigten Komponenten darstellt. Die Einschränkungen an die Anweisungsstruktur werden in den relevanten Kapiteln (Kapitel 3.3 und Kapitel 5.1) im Detail erläutert.

2.2 Schnittstellen zur Definition von Roboteraktionen

In diesem Kapitel werden zunächst die prinzipiellen Methoden beschrieben, welche zur Definition von ausführbaren Roboteraktionen eingesetzt werden können. Anschließend wird das Verfahren im Detail erläutert, welches als Basis für die im Rahmen dieser Arbeit zu entwerfenden Konzepte verwendet wird. Die Einordnung der betrachteten Schnittstelle in das Gesamtsystem ist in Abbildung 2.3 dargestellt.



Abbildung 2.3: Inhaltliche Zuordnung des Kapitels im Gesamtsystem.

Generell kann eine auszuführende Roboteraktion auf verschiedenen Ebenen der Abstraktion definiert werden. Die Ebene der Abstraktion entspricht im Allgemeinen dem Raum, bzw. dem Koordinatensystem (KS) im dem die auszuführende Aktion beschrieben wird. In der Literatur wird typischerweise zwischen folgenden Ebenen unterschieden (vgl. [Zielinski95, Nof99]):

- *Gelenkraum-Ebene* (joint level). Auf dieser Ebene muss zur Definition einer ausführbaren Aktion die erforderliche Bewegung jedes einzelnen Gelenkfreiheitsgrades des Robotersystems festgelegt werden.
- *Arbeitsraum-Ebene* (manipulator level). Auf dieser Ebene wird eine ausführbare Aktion in Bezug auf Größen des Arbeitsraums, bzw. eines globalen Basiskoordinatensystems definiert. Diese Größen können im Allgemeinen in korrespondierende Größen der Gelenkraum-Ebene transformiert werden. Bei kinematischen Größen wird der Zusammenhang durch die inverse Kinematik beschrieben. Bei modernen Robotersystemen mit sechs oder mehr Freiheitsgraden ist diese Abbildung im Allgemeinen nicht eindeutig [Craig05].
- *Aufgabenraum-Ebene* (task level). Auf dieser Ebene wird eine ausführbare Aktion in Bezug auf Größen eines lokalen Koordinatensystems (*task frame*) definiert. Analog zur Arbeitsraum-Ebene, können die Größen des lokalen Koordinatensystems durch entsprechende Methoden in korrespondierende Größen des globalen Koordinatensystems transformiert werden.

Zusätzlich zu den genannten Ebenen wird in der Literatur typischerweise noch eine vierte Ebene der Abstraktion beschrieben, die *Zielsetzungs-* bzw. *Skill-Ebene* (objective level). Bei der Beschreibung dieser Abstraktionsebene sind jedoch deutliche Unterschiede zwischen einzelnen Arbeiten festzustellen. Während in [Zielinski95] auf dieser Ebene klassische Planungssysteme beschrieben werden, beschreibt [Nof99] auf dieser Ebene bereits Systeme, welche komplexe, symbolische Anweisungen wie *spray interior of car door* verarbeiten können.

Die unterschiedlichen Beschreibungen dieser vierten Abstraktionsebene haben jedoch gemeinsam, dass zusätzliche Informationen über die Umgebung benötigt werden, da nicht alle zur Ausführung benötigten Parameter explizit definiert werden. Je nach gewählter Abstraktion der Aktionsbeschreibung benötigt das System eine gegebenenfalls eine komplexe Transformationschicht. Da hierzu bereits entsprechende Funktionalität im System hinterlegt sein muss, ist diese Ebene im Rahmen dieser Arbeit nicht als Basis zur Definition der ausführbaren Roboteraktionen geeignet. Vielmehr können die im Rahmen dieser Arbeit umgesetzten Konzepte dafür genutzt werden, dem Robotersystem entsprechende Fähigkeiten zur Verfügung zu stellen.

Daher wird eine Schnittstelle ausgewählt, welche eine Definition von ausführbaren Roboteraktionen im Rahmen eines aufgabenspezifischen Koordinatensystems erlauben. In diesem Kontext wird zunächst das Konzept des *Task Frame* (TF) und des *Task Frame Formalismus* (TFF) erläutert [Mason79, Ballard84, DeSchutter88]. Der TF beschreibt ein lokales Koordinatensystem, wohingegen der TFF zusätzlich für jeden Freiheitsgrad des TF eine Menge an Parameter definiert. Diese sind beispielsweise die Regelstrategie und der Sollwert des entsprechenden Freiheitsgrads. Daher wird der TFF typischerweise zur Beschreibung von *sensorbasierten* Bewegungen verwendet. Diese können unterteilt werden in *sensorüberwachte* und *sensorgeführte* Bewegungen [Bruyninckx96]. In Abbildung 2.4 sind Beispiele für die Beschreibung der beiden Ausprägungen von sensorbasierten Bewegungen dargestellt.

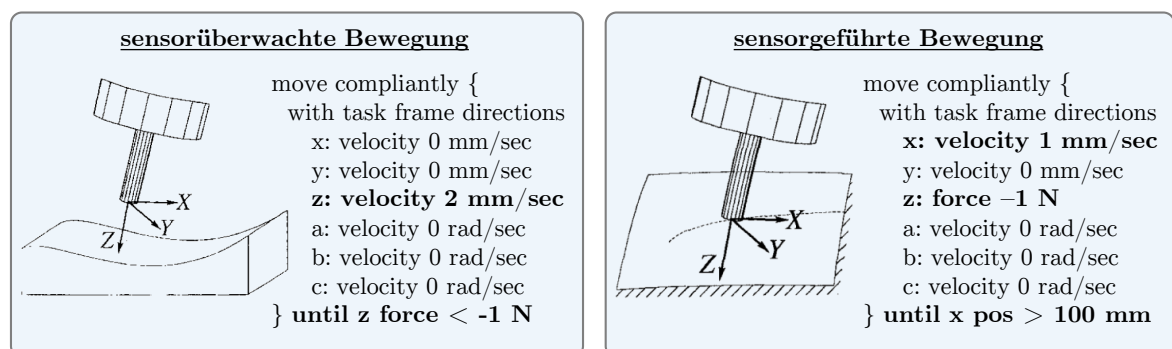


Abbildung 2.4: Beispiele zur Definition von sensorbasierten Roboteraktionen gemäß des *Task Frame Formalismus* (TFF) nach [Bruyninckx96]. Links ist die Definition einer sensorüberwachten Bewegung, rechts die Definition einer sensorgeführten Bewegung dargestellt.

Als Beispiel für eine sensorüberwachte Bewegung ist der Übergang von einer Grobbewegung zu einer Feinbewegung dargestellt [Hwang92]. An dieser Stelle wird zunächst eine Bewegung im Freiraum ausgeführt, welche mit einer konstanten Geschwindigkeit entlang der z-Achse des TF erfolgen soll. Der TF wird für diese Aufgabe am Ende des Werkzeugs festgelegt. Die Bewegung soll so lange durchgeführt werden, bis ein Kontakt zwischen Robotersystem und Umgebung erfolgt. Die Kontaktsituation wird hier durch eine sensorisch erfassbare Kontaktkraft beschrieben, was im TFF durch Angabe eines Abbruchkriteriums formuliert wird. Als Bewegungsparameter werden ausschließlich die Parameter für die Bewegung der z-Achse gesetzt, die Werte der weiteren Freiheitsgrade werden fixiert. Das heißt es sind keine Positions- oder Orientierungsänderungen für die entsprechenden Freiheitsgrade bei der Aktionsausführung erlaubt.

Als Beispiel für eine sensorgeführte Bewegung wird eine kraftgeregelte Feinbewegung entlang einer Oberfläche dargestellt. Wie im Beispiel zur sensorüberwachten Bewegung wird auch hier der TF am Ende des Werkzeugs festgelegt. Zunächst wird eine Bewegung in Richtung der x-Achse des TF mit konstanter Geschwindigkeit definiert. Im Gegensatz zur sensorüberwachten Bewegung soll hier jedoch eine vorgegebene Kontaktkraft in Richtung der z-Achse des TF eingehalten werden. Daher ist für diese Bewegung die z-Achse des TF als kraftgeregelt zu spezifizieren. Für die weiteren Freiheitsgrade ist keine Änderung erlaubt, das heißt diese Werte werden entsprechend fixiert. Die Bewegung im Beispiel soll so lange ausgeführt werden, bis eine entsprechende Richtungsänderung in Richtung der x-Achse des TF durchgeführt wurde, welche die Definition einer entsprechenden Abbruchbedingung erfordert.

Anhand der Beispiele wird gezeigt, dass im Rahmen des TFF eine leistungsfähige Schnittstelle vorhanden ist, welche die einfache und intuitive Definition von sensorbasierten Roboterbewegungen erlaubt. Die praktische Umsetzung und Anwendung der vorgeschlagenen Konzepte erfordert jedoch entsprechend komplexe Roboterhardware, welche beispielsweise eine hybride Kraft- und Positionsregelung erlauben. Daher wurde eine Aktionsdefinition gemäß des TFF zunächst typischerweise für theoretische Betrachtungen verwendet [Kroeger04]. Zudem ist der TFF ausschließlich zur Beschreibung von Roboterbewegungen geeignet. Soll beispielsweise eine Manipulation eines Objekts und die eventuell verbundene Nutzung eines Werkzeugs (zum Beispiel in Form eines Greifers) beschrieben werden, so ist dies im Rahmen des TFF nicht möglich.

Mit der Weiterentwicklung der Roboterhardware und den damit verbundenen Systemen zur Robotersteuerung existieren heutzutage Schnittstellen, welche das Konzept des TFF aufgreifen für die Praxis nutzbar machen [Mosemann01, Bruyninckx03, Thomas03, Finkemeyer04]. Eine der verfügbaren Varianten stellen *Aktionsprimitive* (manipulation primitives), beziehungsweise *Aktionsprimitivnetze* (manipulation primitive nets) dar. Diese wurden am Institut für Robotik und Prozessinformatik (IRP) der Technischen Universität Braunschweig entwickelt [IRP16].

Das Konzept der Aktionsprimitive greift die grundsätzlichen Ideen des TFF auf, erweitert diese an einigen Stellen und stellt eine formale Beschreibung der Schnittstelle zur Verfügung. Eine Erweiterung beschreibt beispielsweise die Nutzung von Werkzeugen während der Bewegungsausführung. Daher ist dieses Konzept nicht nur in der Lage sensorbasierte Roboterbewegungen zu definieren, es stellt eine Schnittstelle zur Verfügung, welche eine allgemeine Beschreibung von werkzeugabhängigen, sensorbasierten Roboteraktionen erlaubt.

Im Rahmen dieser Arbeit muss eine geeignete Schnittstelle zwischen der Transformationsschicht und Kontrollschicht in der Lage sein, sensorbasierte Roboteraktionen systematisch zu beschreiben. Das Konzept der Aktionsprimitive stellt hierzu eine formal definierte Schnittstelle auf Basis der Aufgabenraum-Ebene zur Verfügung. Diese Art der Abstraktion erscheint im Rahmen dieser Arbeit geeignet, da die Semantik einer auszuführenden Aktion gemäß einer lokalen Beschreibung im System hinterlegt werden und somit unabhängig von einer konkreten Roboterumgebung beschrieben werden kann. Daher werden Aktionsprimitive, beziehungsweise Aktionsprimitivnetze als Schnittstelle zwischen Transformation- und Kontrollschicht verwendet.

Im Folgenden werden die einzelnen Bestandteile eines Aktionsprimitivs, beziehungsweise eines Aktionsprimitivnetzes beschrieben (vgl. [Finkemeyer04, Thomas08]). Ein *Aktionsprimitiv* (manipulation primitive) ist definiert als ein 3-Tupel der Form:

$$\mathcal{MP} := (\mathcal{HM}, \tau, \lambda) \quad (2.1)$$

Die einzelnen Komponenten eines Aktionsprimitivs \mathcal{MP} sind

- die *Hybride Bewegung* (hybrid motion) \mathcal{HM} ,
- die Menge der *Werkzeugkommandos* (tool commands) τ und
- die Menge der *Abbruchkriterien* (stopping criteria) λ .

Die hybride Bewegung \mathcal{HM} beschreibt die auszuführende Bewegung basierend auf den TFF. Daher enthält diese Komponente den zur Ausführung der sensorbasierten Bewegung festgelegten Task Frame \mathcal{TF} , dessen Lage \mathcal{P} in Bezug auf ein Referenzkoordinatensystem \mathcal{RF} spezifiziert wird. Die zweite Komponente stellt die adaptive Selektionsmatrix \mathcal{D} dar, welche den Sollwert v_i und Regelstrategie t_i jedes einzelnen Freiheitsgrads von \mathcal{TF} beschreibt. Die formale Definition der Komponenten lautet wie folgt:

$$\mathcal{HM} := (\mathcal{TF}, \mathcal{D}) \quad (2.2)$$

$$\mathcal{TF} := (\mathcal{P}, \mathcal{RF}) \quad (2.3)$$

$$\mathcal{P} = (p_x, p_y, p_z, p_a, p_b, p_c)^T \in \mathbb{R}^6 \quad (2.4)$$

$$\mathcal{RF} = (rf_x, rf_y, rf_z, rf_a, rf_b, rf_c)^T \in \mathbb{R}^6 \quad (2.5)$$

$$\mathcal{D} := \{(v_i, t_i) | i \in 1, \dots, 6\} \quad (2.6)$$

$$v_i \in \mathbb{R}, t_i \in \{\text{Strategies}\}$$

Die Menge der Werkzeugkommandos τ beschreibt die während der Ausführung eines Aktionsprimitivs aktiven Werkzeugparameter. Im Rahmen des Formalismus werden die Werkzeugkommandos in Form von Tupeln beschrieben, in dem sowohl das Werkzeug w_j als auch die Funktion des Werkzeugs c_j durch einen symbolischen Namen beschrieben wird. So kann beispielsweise das Tupel (*Greifer, offen*) auf einen während der Ausführung des Aktionsprimitiv geöffneten Greifers hinweisen. Die formale Definition der Komponente lautet wie folgt:

$$\tau := \{(w_j, c_j) | w_j \in \{Tools\}, c_j \in \{Commands\}\} \quad (2.7)$$

Die Menge der Abbruchkriterien λ beschreiben die Bedingungen, welche den Zielzustand eines Aktionsprimitivs beschreiben. Hierbei wird unterschieden zwischen Default- und Benutzerkriterien. Die Basis für die Abbruchkriterien stellen die verfügbaren Sensorsignale \mathcal{S} dar. Eine Abbruchbedingung λ_i ist im Allgemeinen eine Funktion, welche die Sensorsignale auf einen booleschen Wert abbildet. In der Regel handelt es sich hierbei um logische Ausdrücke, welche ein Sensorsignal mit Konstanten oder Funktionswerten vergleichen. Die formale Definition der Komponente lautet wie folgt:

$$\lambda_i : \mathcal{S} \rightarrow \{true, false\} \quad (2.8)$$

Bei einem Aktionsprimitivnetz handelt es sich um einen gerichteten Graphen, welcher Aktionsprimitive MP als Knoten und Abbruchkriterien von Aktionsprimitiven als Kanten K enthält. Zusätzlich wird eine Menge für Start- und Endknoten MP_{start} , bzw. MP_{stop} festgelegt. Die formale Definition lautet wie folgt:

$$MPN := (MP, MP_{start}, MP_{stop}, K) \quad (2.9)$$

$$MP \in \mathcal{MP}^n, n \in \mathbb{N}$$

$$MP_{start}, MP_{stop} \subseteq MP$$

$$K := \{(MP_i, MP_j, \lambda_{i,k})\}$$

$$MP_i, MP_j \in MP, \quad i, j \in 1, \dots, n \quad \lambda_{i,k} \in \lambda_i$$

Abschließend lässt sich festhalten, dass Roboteraktionen auf verschiedenen Ebenen der Abstraktion beschrieben werden können. Während Beschreibungen der Gelenkraum-Ebene abhängig vom aktuellen Robotersystem, beziehungsweise der aktuellen Roboterkinematik sind, können die Beschreibungen der Arbeitsraum- und Aufgabenraum-Ebene ausführbare Aktionen unabhängig von der aktuellen Roboterkinematik beschreiben. Durch die Verwendung von Beschreibungen auf Aufgabenraum-Ebene können ausführbare Aktionen zudem prinzipiell unabhängig von der aktuellen Umgebung des Robotersystems beschrieben werden. Da im Rahmen dieser Arbeit die Semantik von Symbolen möglichst allgemein beschrieben werden soll, ist eine Aktionsbeschreibung auf Basis der Aufgabenraum-Ebene denen anderer Klassen vorzuziehen. Im Rahmen von Aktionsprimitiven, beziehungsweise Aktionsprimitivnetzen steht eine solche Schnittstelle zur Verfügung. Daher wird diese als Schnittstelle zwischen der zu entwerfenden Transformationsschicht und der Kontrollschicht des Robotersystems festgelegt.

2.3 Gesamtsysteme zur intuitiven Roboterkommandierung

Nachdem in den vorangegangenen Kapiteln die prinzipiellen Methoden zur Umsetzung der Benutzer- und Kontrollschicht erläutert wurden, werden im Folgenden die Umsetzung der Transformationsschicht sowie die bereits vorhandenen Gesamtsysteme betrachtet (siehe Abbildung 2.5). Dies umfasst eine grobe Kategorisierung der Gesamtsysteme anhand der gewählten symbolischen Eingabe und subsymbolischen Ausgabe, einer Analyse der umgesetzten Transformationsschicht sowie einer abschließenden Bewertung anhand geeigneter Merkmale.

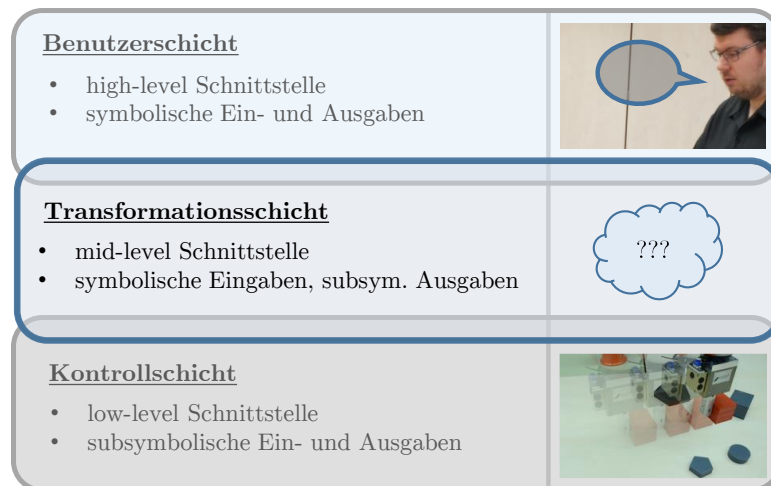


Abbildung 2.5: Inhaltliche Zuordnung des Kapitels im Gesamtsystem.

Kategorisierung anhand der umgesetzten Benutzer- und Kontrollschicht

Die Kategorisierung der vorhandenen Systeme wird zunächst anhand der umgesetzten Benutzer- und Kontrollschicht durchgeführt. Für die Beschreibung der umgesetzten Benutzerschicht wird die Art der *Interaktionsschnittstelle* und für die Klassifikation der Kontrollschicht wird die Art der ausführbaren *Bewegung* verwendet. Die Ausprägungen der beiden Merkmale werden im Folgenden beschrieben, zuvor werden die betrachteten Systeme eingegrenzt.

Im Rahmen der Übersicht werden ausschließlich Arbeiten betrachtet, welche eine symbolische Benutzerschnittstelle basierend auf einer textuellen Repräsentation der Eingabe umsetzen. Dies umfasst daher zum einen die Systeme, welche eine direkte textuelle Eingabe einer auszuführenden Roboteraktion zulassen. Zum anderen umfasst dies die Systeme, welche eine Spracheingabe verwenden und diese zunächst in eine textuelle Repräsentation umwandeln. Zudem werden ausschließlich die Arbeiten betrachtet, welche eine intuitive Benutzerschnittstelle basierend auf den natürlichsprachlichen Symbolen umsetzen. Das heißt die betrachteten Systeme erlauben eine Eingabe in Form konkreter Worte, welche im allgemeinen Sprachverbrauch verwendet werden und keine abstrakten Beschreibungen, welche entsprechendes Expertenwissen voraussetzen. Daher werden beispielsweise keine symbolischen Planungssysteme oder textuelle Programmiersprachen betrachtet.

Die Art der Interaktionsschnittstelle wird unterteilt in *strukturiert* und *semi-strukturiert*. Die in Kapitel 2.1 beschriebene dritte Ausprägung von textuellen Repräsentationen (unstrukturiert) wird im Rahmen der Kategorisierung nicht verwendet. Dies ist zulässig, da jedes der betrachteten Systeme ausschließlich sinnvolle Eingaben gemäß einer Grammatik der verwendeten Eingabesprache erlaubt. Daher wird von keinem der betrachteten Systeme eine unstrukturierte, textuelle Repräsentation umgesetzt.

Die Art der ausführbaren Bewegung wird in *Grobbewegungen* und *Feinbewegungen* unterteilt. Wie im Kapitel 2.2 dargestellt, umfassen die Grobbewegungen typischerweise sensorüberwachte, positionsgeregelte Roboterbewegungen im Freiraum und die Feinbewegungen typischerweise sensorgeführte, hybridgeregelte Roboterbewegungen in Kontaktsituationen (vgl. Abbildung 2.4).

		Art der Bewegung	
		Grobbewegung	Feinbewegung
Art der Interaktions-schnittstelle	strukturiert	[Nilsson84] [Sato87] [Torrance94] [Lauria02] [Pires05]	[Laengle95] [Knoll97]
	semi-strukturiert	[Kemke07] [Kress07] [Rybski08] [Dzifcak09] [Kollar10] [Tellex11] [Thomas12] [Matuszek13] [Kollar14] [Yang14] [Misra15] [Jia16]	[Tenorth10] [Bollini13] [Stenmark15] [SpangenbergXX]

Tabelle 2.2: Klassifikation der eigenen und verwandten Arbeiten auf Basis der umgesetzten Benutzer- und Kontrollschicht. Für die einzelnen Gesamtsysteme wurde jeweils eine stellvertretende Publikation ausgewählt.

Somit lassen sich die vorhandenen Gesamtsysteme auf Basis der gewählten Merkmale in vier Gruppen kategorisieren. Eine tabellarische Übersicht der klassifizierten Gesamtsysteme ist in Tabelle 2.2 dargestellt. Im Folgenden werden die einzelnen Kategorien von Arbeiten beschrieben. Hierzu werden zunächst die prinzipiellen Komponenten und Eigenschaften der Systeme in einer allgemeinen Form dargestellt. Dies umfasst eine Beschreibung der Komponenten, welche alle vier Gruppen von Gesamtsystemen besitzen, sowie der Beschreibung der zusätzlichen Komponenten, welche charakteristisch für die umgesetzte Funktionalität einer entsprechenden Gruppe sind (siehe Abbildungen 2.6 - 2.9). Zur Zuordnung der betrachteten Gruppe ist diese im rechten oberen Bereich einer entsprechenden Abbildung dargestellt.

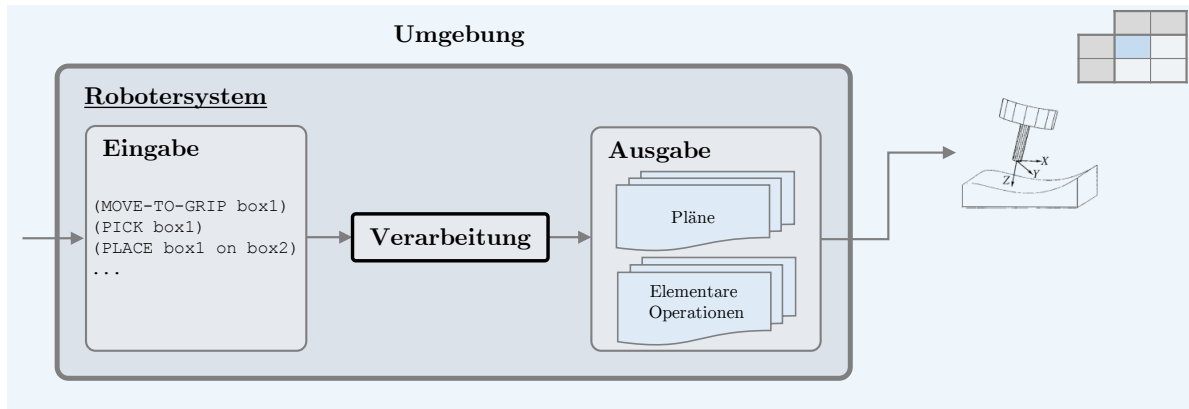


Abbildung 2.6: Prinzipielle Komponenten eines Robotersystems, welches die Verarbeitung von strukturierten Eingaben und Ausführung von Grobbewegungen erlaubt.

Die erste Gruppe von Arbeiten beschreibt Systeme, welche die Verarbeitung von *strukturierten* Eingaben und Ausführung von *Grobbewegungen* erlauben. Die prinzipiellen Komponenten solcher Systeme sind in Abbildung 2.6 dargestellt. Bei diesen Systemen werden im Allgemeinen vollständig bekannte Anweisungen gemäß einer spezifizierten Syntax instruiert. Daher wird die Abbildung zwischen einer Eingabe und der damit verbundenen Ausführung durch das Robotersystem typischerweise im Vorfeld im System hinterlegt. Die auszuführenden Aktionen werden im Allgemeinen in Form eines Plans beschrieben, welcher auf einer Reihe an elementaren Operationen basiert. Für den Bereich der Manipulation sind dies typischerweise Aktionen zum Verfahren des Robotersystems oder zum Greifen und Ablegen eines bekannten Objekts.

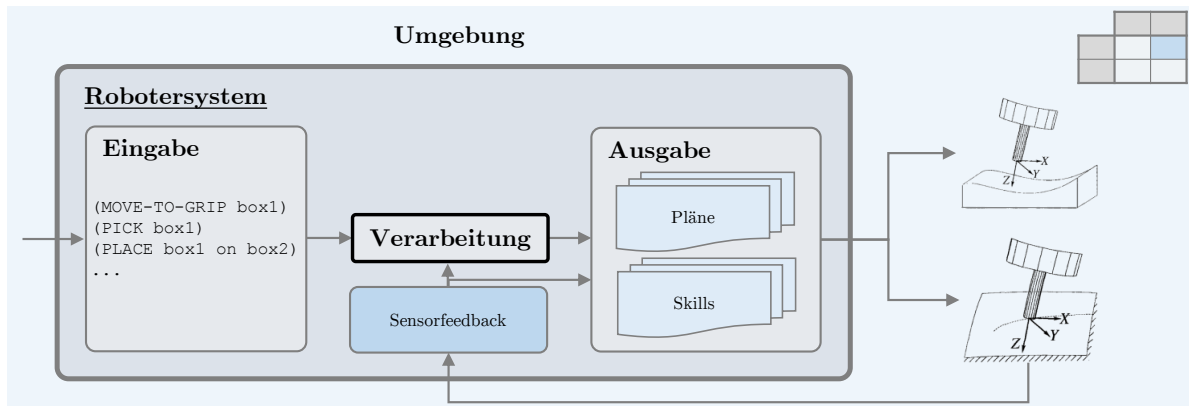


Abbildung 2.7: Prinzipielle Komponenten eines Robotersystems, welches die Verarbeitung von strukturierten Eingaben und Ausführung von Feinbewegungen erlaubt.

Die zweite Gruppe von Arbeiten beschreibt Systeme, welche die Verarbeitung von *strukturierten* Eingaben und Ausführung von *Feinbewegungen* erlauben. Die prinzipiellen Komponenten solcher Systeme sind in Abbildung 2.7 dargestellt. Wie auch bei der ersten Gruppe werden bei diesen Systemen typischerweise vollständig bekannte Anweisungen durch das Robotersystem ausgeführt. Daher ist der prinzipielle Aufbau dieser Systeme zunächst vergleichbar. Da bei diesen Systemen zusätzlich zu den Grobbewegungen sensorbasierte Feinbewegungen ausgeführt werden sollen, benötigen diese Systeme zusätzliche Komponenten.

Zur Ausführung einer sensorbasierten Bewegung wird die sensorische Rückmeldung zur Laufzeit benötigt. Daher verfügen diese Systeme über entsprechende Sensorik und Regelkomponenten, welche die Ausführung überwachen, beziehungsweise bei Abweichung der Sollwerte aktiv in Form einer Regelung in die Ausführung eingreifen. Die Abbildung einer Eingabe erfolgt auch hier im Allgemeinen direkt und in Form von im Vorfeld im System hinterlegter Pläne. Diese basieren ebenfalls auf elementaren Operationen, welche im Kontext von Feinbewegungen typischerweise als *Skills* bezeichnet werden.

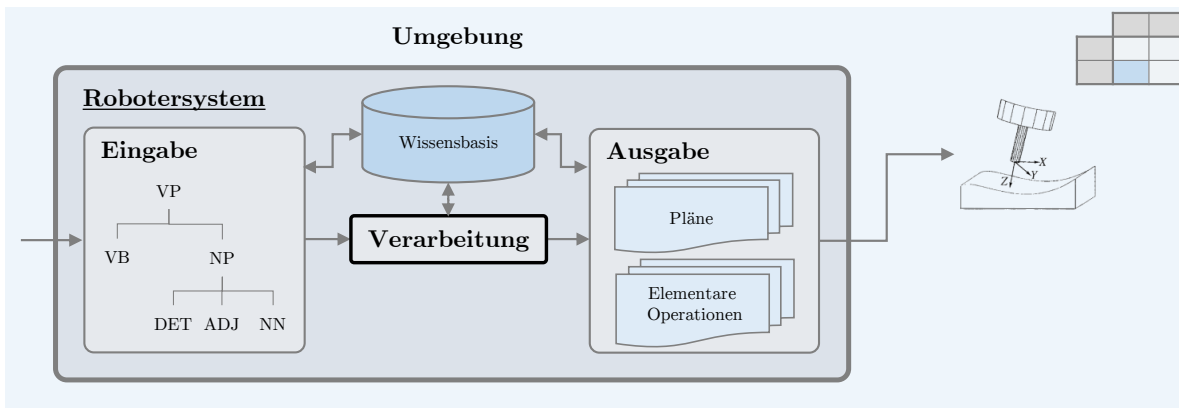


Abbildung 2.8: Prinzipielle Komponenten eines Robotersystems, welches die Verarbeitung von semi-strukturierten Eingaben und Ausführung von Grobbewegungen erlaubt.

Die dritte Gruppe von Arbeiten beschreibt Systeme, welche die Verarbeitung von *semi-strukturierten* Eingaben und Ausführung von *Grobbewegungen* erlauben. Die prinzipiellen Komponenten solcher Systeme sind in Abbildung 2.8 dargestellt. Im Gegensatz zu den Systemen den zuvor beschriebenen Gruppen wird bei diesen Systemen keine konkrete Syntax der auszuführenden Anweisung vorgegeben, jedoch eine grundlegende Struktur der Anweisung erwartet. Diese wird typischerweise anhand syntaktischer Funktionen [Bussmann08] festgelegt. Daher benötigen diese Systeme eine zusätzliche Komponente, welche die entsprechende Information zu den Symbolen der Eingabe zur Verfügung stellt. Diese wird im Allgemeinen als Wissensbasis bezeichnet und kann je nach Ausprägung, beziehungsweise Komplexität einen Einfluss auf die Komponenten der Eingabe, Verarbeitung und Ausgabe besitzen.

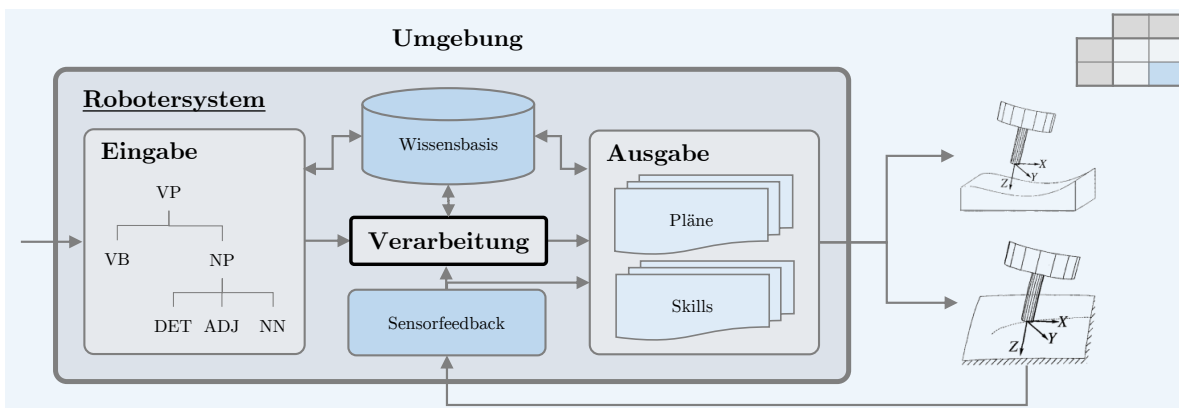


Abbildung 2.9: Prinzipielle Komponenten eines Robotersystems, welches die Verarbeitung von semi-strukturierten Eingaben und Ausführung von Feinbewegungen erlaubt.

Die vierte Gruppe von Arbeiten beschreibt Systeme, welche die Verarbeitung von *semi-strukturierten* Eingaben und Ausführung von *Feinbewegungen* erlauben. Die prinzipiellen Komponenten solcher Systeme sind in Abbildung 2.9 dargestellt. Da diese Systeme sowohl eine semi-strukturierte Eingabe als auch die Ausführung von Feinbewegungen erlauben, werden hier die entsprechenden sensorischen Komponenten als auch eine Wissensbasis benötigt. In diese Gruppe kann das im Rahmen dieser Arbeit umgesetzte System eingruppiert werden.

Abschließend lässt sich festhalten, dass die Realisierung eines Systems gemäß einer der vier vorgestellten Gruppen eine Reihe an Komponenten erfordert, welche je nach Zielsetzung des Gesamtsystems entsprechend ausgeprägt sein müssen. Bei der bisherigen Betrachtung sind dies im Wesentlichen eine *Wissensbasis* sowie eine Komponente zum *Sensorfeedback*.

Analyse der umgesetzten Transformationsschicht

Eine Komponente, welche von der Transformationsschicht zur Verarbeitung der Eingaben genutzt wird, stellt die Wissensbasis dar. In dieser ist die Semantik der zur Verfügung stehenden Symbole in einer geeigneten Form hinterlegt, was im Allgemeinen als *Symbol Grounding* [Harnad90] bezeichnet wird. Je nach Art der beschriebenen Symbole wird zwischen dem *Physical* [Vogt02] und *Social Symbol Grounding* [Cangelosi06] unterschieden. Während im Rahmen des Social Symbol Grounding die Erfassung und Verteilung von Symbolen innerhalb einer Population von Agenten untersucht wird, werden im Rahmen des Physical Symbol Grounding die Zusammenhänge zwischen Symbolen und sensorischer Information untersucht [Coradeschi13]. Daher lassen sich alle betrachteten Arbeiten dem Bereich des Physical Symbol Grounding zuordnen.

Betrachtet man die Transformationsschicht, bzw. die Verarbeitung der Eingabe nicht mehr als Blackbox, so lassen sich auch an dieser Stelle eine Reihe an prinzipiellen Komponenten und Strukturen identifizieren, welche alle betrachteten Systeme in individueller Form umsetzen (siehe Abbildung 2.10).

Eine der prinzipiellen Komponenten der Transformationsschicht stellt ein *Parser* dar, welcher die semi-strukturierte Eingabe entgegennimmt und unter Nutzung der Wissensbasis in eine *Zwischenrepräsentation* überführt. Je nach System handelt es sich hier um eine oder mehrere Repräsentationen, welche entweder symbolisch, subsymbolisch oder hybrid beschrieben werden. Die zweite Komponente im Rahmen der Transformationsschicht stellt im Allgemeinen ein *Parametrisierer* dar, welcher die Zwischenrepräsentation auf die Schnittstelle der subsymbolischen Kontrollschicht abbildet und die zur Ausführung benötigten Parameter ermittelt. Im Folgenden werden die betrachteten Systeme auf Basis der umgesetzten Benutzer-, Transformations- und Kontrollschicht kurz beschrieben.

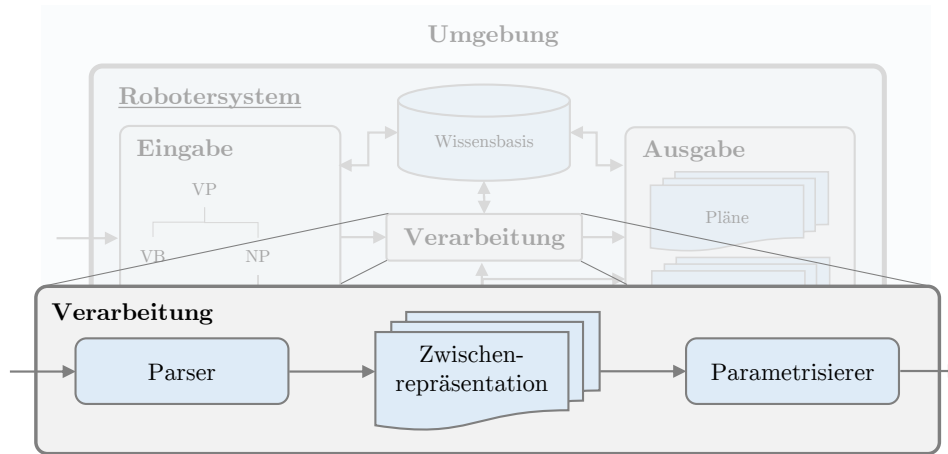


Abbildung 2.10: Prinzipielle Komponenten der Transformationsschicht.

Kurzbeschreibung der Gesamtsysteme

Im Folgenden werden die in Tabelle 2.2 kategorisierten Systeme beschrieben. Die Reihenfolge der Beschreibung wird hierbei gemäß den vier Gruppen vorgenommen. Im Rahmen der Kurzbeschreibung werden zunächst allgemeine Informationen wie beispielsweise der verwendete Robotertyp oder das Anwendungsgebiet beschrieben. Des Weiteren werden die charakteristischen Merkmale wie beispielsweise die möglichen Eingaben, die Verarbeitung, die Zwischenrepräsentationen, die Primitive zur Ausführung und die Parametrisierung der auszuführenden Aktion beschrieben.

Gruppe strukturiert/Grobbewegungen

In [Nilsson84] wird das mobile Robotersystem *Shakey* beschrieben, welches es erlaubt Navigationsaufgaben in Form von vordefinierten *Intermediate-Level-Actions* wie beispielsweise *GOTO point* oder *GOTO room* zu instruieren. Zur Verarbeitung und Ausführung wird das Planungssystem *STRIPS* (STanford Research Institute Problem Solver) [Fikes71] genutzt, welches die auszuführenden Bewegungen anhand eines bekannten Weltmodells berechnet und eine direkte Abbildung in elementare Roboteraktionen vornimmt. Die subsymbolischen Parameter zur Aktionsdefinition sind im Rahmen des bekannten Weltmodells hinterlegt.

In [Sato87] wird das stationäre Robotersystem *LARTS* (Language-Aided Robotic Teleoperation System) beschrieben, welches die textuelle Kommandierung eines Manipulators im Bereich der Teleoperation basierend auf einer Menge an vordefinierten Anweisungen gemäß der *TOL.O* (Teleoperation-Oriented Language of Object-Level) beschreibt. Diese stellt Instruktionen wie beispielsweise (PICK obj), (PLACE obj1 ON obj2) oder (MOVE-TO-GRIP obj) bereit. Diese werden direkt auf eine auszuführende Roboteraktionen abgebildet. Die subsymbolischen Parameter zur Aktionsdefinition werden im Vorfeld ermittelt und im System hinterlegt.

In [Torrance94] wird die multimodale Navigation des mobilen Robotersystems *TJ* beschrieben. Hierzu stehen im System vordefinierte Befehle wie beispielsweise `[Turn] {right|left} [degrees [degrees]]` oder `Go [about] number {inches|feet|yards}` zur Verfügung. Diese werden zunächst in eine Normalform überführt und anschließend durch ein Planungssystem verarbeitet. Ähnlich wie in [Nilsson84] werden hier die benötigten geometrischen Informationen zur Aktionsdefinition im Vorfeld im Rahmen eines Weltmodells hinterlegt.

In den Arbeiten [Lauria02,Kyriacou02,Kyriacou05] wird die Navigation eines mobilen Miniaturroboters innerhalb einer Miniaturstadt beschrieben. Zur Instruktion des Systems werden sogenannte *Primitive Procedures* wie beispielsweise `TAKE THE [number] turn [(left|right)] | [(before|after|at) landmark]` oder `GO (before|after|to) landmark` verwendet. Diese werden im Vorfeld auf eine Reihe von vordefinierten *Sensory-Motor Primitives* abgebildet. Die subsymbolischen Parameter zur Aktionsdefinition sind im Rahmen der Sensory-Motor Primitives hinterlegt.

In [Pires05] wird die Steuerung von mehreren stationären Industrierobotersystemen basierend auf Spracheingaben beschrieben. Hierbei werden die Instruktionen in der Form `Robot number command parameter` instruiert. Jedes Robotersystem besitzt eine eindeutige Nummer und als Kommandos stehen beispielsweise *motor on* oder *program run* zur Verfügung. Die Abbildung der Eingabe auf die subsymbolische Aktionsbeschreibung erfolgt hierbei direkt, subsymbolische Parameter sind im Rahmen von im Vorfeld im System hinterlegten Prozeduren spezifiziert.

Gruppe strukturiert/Feinbewegungen

In den Arbeiten [Lueth94,Laengle95] wird die natürlichsprachliche Steuerung *KANTRA* (KAmro Natural language TRAnslator) des mobilen Robotersystems *KAMRO* (Karlsruhe Autonomous Mobile RObot, [Hoermann91]) beschrieben. Zur Instruktion stehen eine Reihe an elementaren Manipulationsaufgaben zur Verfügung, welche auf unterschiedlichen Abstraktionsebenen dem System übermittelt werden können. Diese kann beispielsweise in Form von zusammengesetzten Aktionssequenzen wie `assemble benchmark` oder elementaren Aktionen wie `grasp obj` oder `move pos` dem System übermittelt werden. Die Ausführung erfolgt durch das Planungssystem *FATE*, welches beispielsweise für die zusammengesetzten Aktionssequenzen eine Reihe an Standardwerten zur Parametrisierung von ausführbaren Roboteraktionen bereitstellt.

In den Arbeiten [Knoll97,Zhang99] wird die Manipulation von bekannten Objekten durch ein stationäres Mehrarmsystem beschrieben. Zur Instruktion können dem System beispielsweise Anweisungen wie `PICK-UP obj` oder `SCREW obj1 INTO obj2` instruiert werden. Diese werden auf eine Reihe im System hinterlegter Roboterfähigkeiten abgebildet. Die Parameter zur Ausführung der sensorbasierten Roboterfähigkeiten werden teilweise automatisiert extrahiert, beispielsweise die geometrischen Parameter. Teilweise werden diese im Vorfeld festgelegt, beispielsweise die dynamischen Parameter.

Gruppe semi-strukturiert/Grobbewegungen

In [Kemke06,Kemke07] wird die Steuerung eines mobilen Systems basierend auf natürlichsprachlichen Eingaben beschrieben. Die Eingaben werden zunächst anhand der syntaktischen Funktionen analysiert und auf hinterlegte Templates wie beispielsweise `bring<obj, loc, dest>` abgebildet. Die Abbildung auf ausführbare Roboteraktionen erfolgt durch ein Planungssystem, welches aus vordefinierten und generischen Aktionen besteht, welche durch geometrische Parameter spezifiziert werden.

In [Rybski07,Rybski08] wird die Navigation eines mobilen Systems basierend auf natürlichsprachlichen Eingaben beschrieben. Diese werden auf sogenannte *Tasks* abgebildet, welche wiederum auf einer Reihe von Verhalten basieren. Der Fokus der Arbeit liegt hierbei auf der Beschreibung einer Dialogstruktur. Die Verhalten, welche eine Bewegung des Roboters zur Folge haben beschränken sich auf `Goto(name)` oder `Follow(person)`. Die zur Ausführung benötigten subsymbolischen Parameter werden entweder einer im System hinterlegten Karte oder einem Trackingsystem entnommen.

In [Dzifcak09] wird die Navigation eines mobilen Systems basierend auf natürlichsprachlichen Eingaben beschrieben. Diese werden zunächst gemäß einer *Combinatorial Categorical Grammar* (CCG) geprüft und in eine Logik-basierte Zwischendarstellung überführt. Die Ausdrücke der Zwischendarstellung werden auf sogenannte *action scripts* abgebildet, welche durch das Robotersystem ausgeführt werden können. Die zur Ausführung benötigten subsymbolischen Parameter werden hierbei im Vorfeld zur Symbolinformation hinterlegt.

Die Arbeiten [Kollar10,Tellex11,Kollar14] beschreiben ebenfalls die natürlichsprachliche Steuerung eines mobilen Systems im Bereich der Navigation. Eine Eingabe wird hier zunächst in sogenannte *spatial description clauses* (SDCs) überführt. Jedes SDC besitzt die Attribute Akteur, Aktion, Ort und Relation. Im System werden eine Reihe an Möglichkeiten für die einzelnen Attribute hinterlegt. Die Zuordnung zwischen SDCs und auszuführenden Aktionen erfolgt hierbei wahrscheinlichkeitsbasiert unter Nutzung einer Menge an low-level Features. Die zur Ausführung benötigten subsymbolischen Parameter werden aus einer im System vorhandenen Karte extrahiert.

In [Thomas12] wird die natürlichsprachliche Kommandierung eines mobilen Robotersystems beschrieben. Eine Eingabe wird zunächst in sogenannte *semantic frames* (SFs) überführt. An dieser Stelle sei darauf hingewiesen, dass es sich hierbei nicht um Frames im Sinne von Koordinatensystemen handelt. Der Begriff stammt aus dem Bereich der Linguistik und beschreibt im Wesentlichen die einzelnen Komponenten (*frame elements*, FEs), welche zur Ausführung einer Szene benötigt werden. Betrachtet man beispielsweise den SF der Aktion *drehen*, so besitzt dieser die FEs Richtung, Distanz und die Einheit der Distanz. Nachdem die Eingabe in SFs überführt wurde, werden diese in einen Plan überführt, welcher auf einer Menge vordefinierter Verhalten basiert. Diese werden im Anschluss durch das Robotersystem ausgeführt.

In [Matuszek13, Matuszek15] wird ebenfalls die natürlichsprachliche Steuerung eines mobilen Robotersystems beschrieben. Die ausführbaren Aktionen werden hierbei zunächst in eine Zwischenrepräsentation gemäß der *Robot Control Language* (RCL) überführt. Anschließend wird die Zwischenrepräsentation an ein Planungssystem weitergegeben und basierend auf einer Menge elementarer Operationen ausgeführt.

In den Arbeiten [Guha13, Yang14, Zampogiannis15] wird die natürlichsprachliche Kommandierung von Aufgaben aus dem Bereich der Manipulation beschrieben. Hierzu wird ein stationäres Zweiarmsystem verwendet, welches zunächst die Eingaben in eine Zwischenrepräsentation gemäß der spezifizierten *Manipulation Action Context-Free Grammar* (MACFG) überführt. Diese Zwischenrepräsentation wird in *Visual Semantic Graphs* (VSGs) überführt, welche anhand vordefinierter Verhalten ausgeführt werden.

In [Misra15] wird die natürlichsprachliche Steuerung eines mobilen Robotersystems beschrieben, welches Aufgaben aus dem Bereich der Navigation als auch Manipulation ausführt. Wie in [Laengle95] können auch hier Anweisungen auf unterschiedlichen Abstraktionsebenen instruiert werden. Die Eingaben werden zunächst in eine formale Darstellung überführt, welche auf einer Menge von low-level Funktionen wie beispielsweise `grasp(obj)`, `moveTo(obj)` oder `release(obj)` basiert. Die zur Ausführung benötigten Parameter sind im Rahmen eines Weltmodells und einer Objektdatenbank verfügbar.

Ein dazu ähnlicher Ansatz basierend auf *Robot Primitive Actions* wird im Rahmen von [Jia16] beschrieben. Hier werden ebenfalls die Eingaben zunächst in eine formale Zwischenrepräsentation überführt und anschließend eine Reihe von low-level Aktionen wie `openG` (open Gripper), `closeG` (close Gripper) und `move(x)` als Basis für auszuführende Aktionen verwendet. Die zur Ausführung benötigten Parameter sind im Rahmen einer Objektdatenbank verfügbar.

Gruppe semi-strukturiert/Feinbewegungen

In [Tenorth10, Tenorth11, Beetz11] wird das Erzeugen eines ausführbaren Plans aus einer Schritt-für-Schritt Anleitung beschrieben, wobei die Eingabe in Form einer Webseite vorliegt. Die darin formulierten Instruktionen werden zunächst in eine formale Zwischenrepräsentation überführt. Anschließend wird diese in einen ausführbaren Plan gemäß der *Reactive Plan Language* (RPL) überführt. Die Ausführung basiert auf im Vorfeld gelernten Aktionsmodellen, die zur Ausführung benötigten Parameter werden teilweise im Rahmen eines Weltmodells automatisch extrahiert, beispielsweise die benötigten geometrischen Parameter. Die dynamischen Parameter werden im Vorfeld im System hinterlegt.

In [Bollini13] wird das Robotersystem *BakeBot* beschrieben, welches in der Lage ist, Backrezepte zu interpretieren und sensorbasiert auszuführen. Die Anweisungen werden hier zunächst in eine strukturierte Darstellung überführt, welche daraufhin in einen Plan übersetzt wird. Dieser basiert auf einer Menge von vordefinierten Aktionen, sogenannte *motion primitives*. Die zur Ausführung benötigten Parameter werden teilweise automatisch extrahiert (geometrische Parameter) und teilweise im Vorfeld spezifiziert (dynamische Parameter).

In [Stenmark15] wird das Instruieren von Industrierobotersystemen basierend auf abstrakten Aufgabenbeschreibungen dargestellt. Die instruierte Anweisung wird hier zunächst unter Nutzung einer Wissensbasis in einen graphbasierten Plan überführt, welcher auf einer iTaSC-basierten Beschreibung der Aktionen (siehe [DeSchutter07]) ausgeführt wird. Die zur Ausführung benötigten Parameter werden in Form von Parametersätzen im System hinterlegt.

Klassifikation und Bewertung

Zum Abschluss dieses Kapitels werden sowohl die eigenen Arbeiten als auch die vorgestellten Systeme klassifiziert und bewertet. Zunächst werden die zur Bewertung verwendeten Kriterien beschrieben. Zum einen sind dies die Merkmale, welche bereits zur groben Klassifikation genutzt werden. Zum anderen sind dies eine Reihe an Merkmalen, welche die umgesetzte Transformationsschicht genauer klassifizieren. In Tabelle 2.3 ist eine tabellarische Übersicht der Klassifikation der eigenen und verwandten Arbeiten dargestellt. Zur Klassifikation werden die folgenden Merkmale mit entsprechenden Ausprägungen verwendet:

- *Interaktionsschnittstelle* (strukturiert/semi-strukturiert).
Dieses Merkmal beschreibt die umgesetzte symbolische Benutzerschnittstelle. Bei einer strukturierten Schnittstelle können ausschließlich Eingaben gemäß einer vorgegebenen Syntax verarbeitet werden, bei einer semi-strukturierten Schnittstelle wird eine gültige Eingabe gemäß der verwendeten Eingabesprache erwartet. Da keine der betrachteten Arbeiten eine unstrukturierte Interaktionsschnittstelle umsetzt wird diese Ausprägung nicht genutzt.
- *Aktionsbeschreibung* (Arbeitsraum/Aufgabenraum).
Dieses Merkmal beschreibt die Abstraktionsebene der verwendeten Aktionsbeschreibung. Eine Beschreibung im Gelenkraum ist abhängig von der verwendeten Roboterkinematik, während Beschreibungen im Arbeits- und Aufgabenraum prinzipiell unabhängig von konkreten Roboterkinematiken sind. Die Beschreibung im Aufgabenraum ist zudem prinzipiell unabhängig von der aktuellen Umgebung des Robotersystems. Da keine der betrachteten Arbeiten eine Aktionsbeschreibung im Gelenkraum verwendet, wird diese Ausprägung nicht genutzt.
- *Parametertypen* (geometrisch/dynamisch).
Dieses Merkmal beschreibt die Typen an Parameter, welche bei der Beschreibung einer ausführbaren Aktion verwendet und spezifiziert werden können.
- *Parameterextraktion* (vordefiniert/automatisch).
Dieses Merkmal beschreibt in wie weit das Robotersystem in der Lage ist, die zur Ausführung benötigten Parameter automatisiert unter Nutzung zusätzlicher Komponenten zu extrahieren.
- *Bewegungsart* (Grobbewegung/Feinbewegung).
Dieses Merkmal beschreibt die Art der ausführbaren Bewegungen und in wie weit das Robotersystem in der Lage ist sensorbasierte Bewegungen auszuführen.

	Interaktions- schnittstelle	Aktions- beschreibung	Parameter- typen	Parameter- extraktion	Bewegungs- art
[Nilsson84]	○	○	○	○	○
[Sato87]	○	○	○	○	○
[Torrance94]	○	○	○	○	○
[Laengle95]	○	○	●	○	●
[Knoll97]	○	○	●	●	●
[Lauria02]	○	○	○	○	○
[Pires05]	○	○	○	○	○
[Kemke07]	●	○	○	●	○
[Rybski08]	●	○	○	●	○
[Dzifcak09]	●	○	○	○	○
[Kollar10]	●	○	○	●	○
[Tenorth10]	●	●	●	●	●
[Tellex11]	●	○	○	●	○
[Thomas12]	●	○	○	●	○
[Matuszek13]	●	○	○	●	○
[Bollini13]	●	○	●	●	●
[Kollar14]	●	○	○	●	○
[Yang14]	●	○	○	●	○
[Stenmark15]	●	●	●	●	●
[Misra15]	●	○	○	●	○
[Jia16]	●	○	○	●	○
[SpangenbergXX]	●	●	●	●	●

Interaktionsschnittstelle:	○ strukturiert	● semi-strukturiert
Aktionsbeschreibung:	○ Arbeitsraum	● Aufgabenraum (TF)
Parametertypen:	○ geometrisch	● dynamisch
Parameterextraktion:	○ vordefiniert	● teilweise automatisch
Bewegungsart:	○ Grobbewegung	● Feinbewegung

Tabelle 2.3: Kategorisierung der eigenen und verwandten Arbeiten gemäß der festgelegten Merkmale. Die Beschreibung der Symbole ist der dargestellten Legende zu entnehmen.

Betrachtet man die Kategorisierung der vorhandenen Systeme, so lassen sich eine Reihe an Schlussfolgerungen ableiten, welche im Folgenden beschrieben werden.

Zunächst lässt sich festhalten, dass keines der betrachteten Systeme eine systematische Beschreibung und kontextabhängige Parametrisierung von sensorbasierten Feinbewegungen erlaubt. Die Systeme, welche in eine gemeinsame Gruppe mit dem im Rahmen dieser Arbeit umgesetzten Gesamtsystem einsortiert werden (siehe [Tenorth10, Bollini13, Stenmark15]), unterscheiden sich in diversen Merkmalen, beispielsweise in der Art der Aktionsbeschreibung oder der Art der Parameterextraktion.

Zum einen sieht beispielsweise keines der vorhanden Systeme eine vollständige, systematische Beschreibung der ausführbaren Aktionen im Aufgabenraum vor. Soll die Semantik von ausführbaren Aktionen jedoch möglichst allgemein beschrieben und im Robotersystem hinterlegt werden, so ist eine Aktionsbeschreibung im Aufgabenraum denen anderer Arten vorzuziehen, da diese prinzipiell unabhängig von der aktuellen Umgebung des Robotersystems formuliert werden kann.

Zum anderen werden beispielsweise bei den verwandten Systemen die zur Ausführung von sensorgeführten Feinbewegungen benötigten Parameter typischerweise im Vorfeld im System hinterlegt oder müssen explizit durch den Benutzer instruiert werden. Eine automatisierte, kontextabhängige Extraktion durch das Robotersystem ist bei den betrachteten Arbeiten nicht vorgesehen.

2.4 Zusammenfassung

Nachdem die an die Transformationsschicht angrenzenden Schnittstellen der symbolischen Benutzerschnittstelle (siehe Kapitel 2.1), der subsymbolischen Schnittstelle hin zum Robotersystem (siehe Kapitel 2.2), sowie die verwandten Arbeiten vorgestellt und kategorisiert wurden (siehe Kapitel 2.3), werden im Folgenden die gewonnenen Erkenntnisse und die daraus extrahierten Schlussfolgerungen beschrieben.

Im Bereich der symbolischen Schnittstellen lassen sich eine Reihe an Möglichkeiten feststellen, welche die Umsetzung einer symbolischen Benutzerschnittstelle erlauben. In erster Linie werden hierfür das auditive, taktile und visuelle Wahrnehmungssystem des Robotersystems verwendet. Die Eingabe durch den Benutzer erfolgt hierbei im Allgemeinen in Form von Sprachanweisungen, Texteingaben oder Gesten. Diese werden zur Weiterverarbeitung typischerweise in eine textuelle Repräsentation überführt, welche je nach Verfahren entweder strukturiert, semi-strukturiert oder unstrukturiert vorliegen kann. Während strukturierte Eingaben starke Einschränkungen vornehmen, werden an unstrukturierte Eingaben keinerlei Einschränkungen gestellt, was unter Umständen auch zu ungewollten Anweisungen führen kann. Daher stellen semi-strukturierte Eingaben einen Kompromiss zwischen Flexibilität für den Benutzer und Komplexität der Eingabeverarbeitung dar. Im Rahmen dieser Arbeit wird daher festgelegt, die vom Benutzer instruierten Kommandos in einer semi-strukturierten Form entgegenzunehmen.

Im Bereich der subsymbolischen Schnittstellen lassen sich drei grundsätzliche Methoden zur Definition von Roboteraktionen beschreiben. Bei einer Beschreibung auf Gelenkraum-Ebene wird die Bewegung für jeden Gelenkfreiheitsgrad des Robotersystems explizit beschrieben, wodurch diese Art der Beschreibung abhängig von der aktuellen Roboterkinematik ist. Die Beschreibungen auf Arbeits- und Aufgabenraum-Ebene sind prinzipiell unabhängig von der Roboterkinematik, die Beschreibung im Aufgabenraum sogar prinzipiell unabhängig von der aktuellen Umgebung des Robotersystems. Da im Rahmen dieser Arbeit eine flexible und systematische Beschreibung von ausführbaren Roboteraktionen angestrebt wird, wird eine subsymbolische Beschreibung auf Aufgabenraum-Ebene verwendet. Eine solche Schnittstelle steht im Rahmen von Aktionsprimitiven, bzw. Aktionsprimitivnetzen zur Verfügung, welche daher als Schnittstelle zwischen der Kontrollschicht und der zu spezifizierenden Transformationsschicht festgelegt wird.

Im Bereich der Gesamtsysteme lassen sich ebenfalls eine Reihe an Schlussfolgerungen extrahieren. Zunächst ist anhand der groben Kategorisierung die historische Entwicklung und die steigende Komplexität der umgesetzten Systeme erkennbar (siehe Tabelle 2.2). Durch entsprechende Fortschritte im Bereich der Komponenten zur Sprachverarbeitung sind nahezu alle heutigen Systeme in der Lage, symbolische Benutzerschnittstellen anzubieten, welche es erlauben Eingaben in semi-strukturierter Form entgegenzunehmen und zu verarbeiten. Obwohl es auch entsprechende Fortschritte im Bereich der subsymbolischen Beschreibungen von Roboteraktionen als auch in der Verfügbarkeit von hybrid regelbaren Robotersystemen gibt, verwenden viele der aktuellen Systeme jedoch ausschließlich Grobbewegungen zur Ausführung der instruierten Anweisungen.

Betrachten die Systeme das Anwendungsgebiet der Navigation, so liegt der Fokus typischerweise auf dem Verständnis von symbolischen Ortsangaben, zudem ist es bei der Ausführung von Navigationsaufgaben im Allgemeinen das Ziel, einen möglichst großen Abstand zu Hindernissen einzuhalten, daher werden hier typischerweise keine Feinbewegungen benötigt.

Betrachten die Systeme jedoch das Anwendungsgebiet der Manipulation, beziehungsweise Anwendungsdomänen wie Privathaushalte oder kleine und mittelständische Unternehmen, so stellt sich die Frage, warum auch an dieser Stelle typischerweise ausschließlich Grobbewegungen zur Ausführung verwendet werden. Viele der in diesen Bereichen auszuführenden Aufgaben können durch Verwendung von sensorbasierten Feinbewegungen robuster und flexibler beschrieben werden (Beispiele siehe Abbildung 2.11).

Für das Beispiel aus dem Anwendungsbereich Catering kann beispielsweise das Drehmoment am Endeffektor überwacht werden, um festzustellen, ob Kaffee ausgeschenkt wird. Für das Beispiel aus dem Anwendungsbereich Werkstatt kann beispielsweise eine konstante Kontaktkraft normal zum Werkstück vorgegeben werden, um ein gleichmäßiges Ergebnis zu erzielen.



Abbildung 2.11: Anwendungsgebiete zukünftiger Robotersysteme, bei denen sensorbasierte Bewegungen vorteilhaft eingesetzt werden können. Links eine Anwendung aus dem Bereich Catering, rechts eine Anwendung aus dem Bereich Werkstatt.

Ein Grund für die vergleichsweise wenig verbreitete Nutzung von Feinbewegungen bei der Aktionsausführung kann beispielsweise die Anzahl der festzulegenden Parameter darstellen, welche zur Definition einer Feinbewegung benötigt werden. Während bei einer Grobbewegung prinzipiell eine Angabe der Zielposition ausreichend ist (vgl. exemplarisch die Aktion `move pos` aus [Lauria02]), müssen für eine Feinbewegung eine Reihe an Parameter spezifiziert werden (vgl. Definition eines Aktionsprimitivs, Gleichung 2.1).

Einen weiteren Grund für die vergleichsweise wenig verbreitete Nutzung von Feinbewegungen kann eine fehlende systematische Beschreibung der Zusammenhänge zwischen symbolischen Anweisungen, einer zugrunde liegenden subsymbolischen Aktionsbeschreibung und der Parametrisierung einer auszuführenden, sensorbasierten Roboteraktion darstellen.

Die Kombination einer intuitiven Benutzerschnittstelle mit einer leistungsfähigen, sensorbasierten Ausführung stellt jedoch eine wichtige Komponente für die Expansion der Einsatzgebiete zukünftiger Robotersysteme dar [SPARC15]. Daher werden im Rahmen der folgenden Kapitel eine Reihe an Konzepten vorgeschlagen, welche eine systematische Beschreibung der genannten Zusammenhänge zum Ziel haben.

3 Beschreibung von ausführbaren Roboteraktionen

Wie in Kapitel 1.1 dargestellt, sollen die Einsatzgebiete von zukünftigen Robotersystemen auf Anwendungsbereiche wie private Haushalte erweitert werden. Es wurde erkannt, dass hierfür Robotersysteme benötigt werden, welche sowohl eine intuitiv zu verwendende Benutzerschnittstelle zur Verfügung stellen, als auch die Ausführung von sensorbasierten Bewegungen erlauben. Zudem wurde Forschungsbedarf für die Umsetzung von Konzepten formuliert, welche die Zusammenhänge zwischen einer symbolischen Benutzerschicht und einer subsymbolischen Kontrollschicht beschreiben.

Ein Teil stellt hierbei die systematische Beschreibung der symbolisch instruierten Roboteraktionen auf Basis der gewählten subsymbolischen Kontrollschicht dar. Im Rahmen dieses Kapitels wird daher zunächst die wissenschaftliche Fragestellung **F1** untersucht, welche die systematische Beschreibung von ausführbaren Roboteraktionen adressiert und wie folgt lautet:

- F1** In welcher Form kann die Semantik einer ausführbaren Aktion systematisch beschrieben und in einem Robotersystem hinterlegt werden?

Abschließend wird die Struktur dieses Kapitels beschrieben. Zunächst wird die Grundidee des im Rahmen dieser Arbeit entwickelten Ansatzes zur Umsetzung einer Transformationsschicht skizziert (Kapitel 3.1). Anschließend werden die Zusammenhänge zwischen dem vorgeschlagenen Ansatz und der im Rahmen dieser Arbeit verwendeten subsymbolischen Schnittstelle beschrieben (siehe Kapitel 3.2). Wie in Kapitel 2.2 dargestellt, werden Aktionsprimitivnetze als subsymbolische Schnittstelle hin zum Robotersystem verwendet.

Daraufhin werden die Zusammenhänge zwischen dem vorgeschlagenen Ansatz und der im Rahmen dieser Arbeit verwendeten symbolischen Schnittstelle beschrieben (Kapitel 3.3). Wie in Kapitel 2.1 dargestellt, wird als Eingabe semi-strukturierter Text unter Nutzung natürlichsprachlicher Symbole verwendet. Abschließend werden die gewonnenen Erkenntnisse zusammengefasst und diskutiert (Kapitel 3.4).

Die im Folgenden beschriebenen Konzepte sind in [Spangenberg14] und [Spangenberg15] veröffentlicht. In [Spangenberg14] wird der Grundansatz sowie die Zusammenhänge zwischen Benutzer- und Transformationsschicht beschrieben. In [Spangenberg15] werden die Zusammenhänge zwischen Transformations- und Kontrollschicht dargestellt.

3.1 Ansatz

Im Rahmen von symbolischen Anweisungen wie beispielsweise

“Bewege den roten Block gegen den blauen Block!”

“Schiebe den roten Block gegen den blauen Block!”

“Stoße den roten Block gegen den blauen Block!”

repräsentiert typischerweise das Verb der Anweisung die auszuführende Aktion. Werden die genannten Anweisungen einem erwachsenen Menschen übermittelt und die entsprechende Ausführung beobachtet, so lassen sich für alle drei Anweisungen unterschiedliche Ausführungen feststellen [Spangenberg14].

Daraus lässt sich folgern, dass zum einen das verwendete Verb eine wesentliche Rolle bei der Aktionsausführung einnimmt. Dies ist damit zu begründen, dass die weiteren Symbole der genannten Anweisungen identisch gewählt sind. Zum anderen ist der Mensch offensichtlich in der Lage, die Unterschiede zwischen den instruierten Aktionen wahrzunehmen, was sich anhand der unterschiedlichen Ausführung begründen lässt. Somit muss der Mensch eine Vorstellung, beziehungsweise eine interne Beschreibung der Semantik der einzelnen Symbole besitzen. Diese wird im Allgemeinen im Rahmen eines im Kleinkindalter beginnenden, kontinuierlichen Lernprozesses erlangt [Piaget52].

Soll nun ein Robotersystem ebenfalls die Symbolsemantik im Rahmen einer internen Beschreibung besitzen, stellt sich die Frage in welcher Form diese umgesetzt werden kann. Dies erfordert zunächst ein exaktes Verständnis der auszuführenden Aktionen, beziehungsweise ein exaktes Verständnis darüber, in welcher Form sich einzelne Aktionen unterscheiden können.

Im Rahmen dieser Arbeit werden Aktionen aus dem Bereich der Objektmanipulation betrachtet, die diese Gruppe von Aktionen relevante Tätigkeiten beschreiben, welche zukünftige Robotersysteme in Anwendungsdomänen wie beispielsweise privaten Haushalten ausführen sollen (siehe Kapitel 1.3).

Betrachtet man die Ausführung einer solchen Aktion im Detail, so lässt sich diese im Allgemeinen in drei Ausführungsphasen unterteilen. Diese sind die Annäherungsphase, die Manipulationsphase und die Abschlussphase (vgl. [Flanagan06]). Daher unterscheiden sich verschiedene Aktionen, beziehungsweise Verben aus dem Bereich der Objektmanipulation in mindestens einer dieser Ausführungsphasen. Ist dies nicht der Fall, so handelt es sich typischerweise um Verben, welche als Synonyme verwendet werden können.

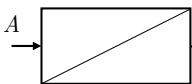
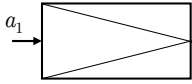
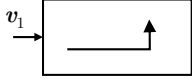
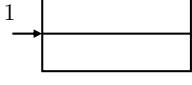

Daher stellt sich die Frage, welche Varianten innerhalb der einzelnen Phasen einer Aufgabe aus dem Bereich der Objektmanipulation existieren und in welcher Form diese beschrieben werden können, sodass die Beschreibung sowohl für eine symbolische als auch für eine subsymbolische Repräsentation geeignet ist.

Im Rahmen dieser Arbeit wird hierzu die Verwendung von physikalischen Effekten, beziehungsweise physikalischen Gesetzmäßigkeiten vorgeschlagen. Diese scheinen geeignet, da die physikalischen Gesetzmäßigkeiten sowohl für die symbolische Repräsentation des Menschen als auch für die subsymbolische Repräsentation des Robotersystem gültig sind und im Allgemeinen nicht außer Kraft gesetzt werden können. Daher wird die Arbeitshypothese formuliert, dass Aufgaben aus dem Bereich der Objektmanipulation in Form von physikalischen Effekten, beziehungsweise physikalischen Gesetzmäßigkeiten und Nebenbedingungen formuliert werden können.

Daraus lassen sich eine Reihe an Fragestellungen ableiten, beispielsweise in welcher Form physikalische Effekte systematisch beschrieben oder in wie weit physikalische Effekte zur Transformation zwischen symbolischen und subsymbolischen Repräsentationen genutzt werden können.

Zur systematischen Beschreibung der physikalischen Effekte wird im Rahmen dieser Arbeit eine Methode aus den Ingenieurwissenschaften adaptiert. Im Bereich der Systementwicklung wird das Konzept der Prinziplösungen zur Beschreibung von diversen Realisierungsmöglichkeiten für technische Problemstellungen verwendet [Koller98]. Hierzu werden die Prinzipien anhand einer Menge elementarer Operationen kategorisiert, entsprechende Prinziplösungen beschrieben und im Rahmen von Konstruktionskatalogen veröffentlicht [Roth00]. In Tabelle 3.1 sind exemplarisch die elementaren Operationen auf Energien, beziehungsweise Energiekomponenten nach [Koller98] dargestellt.

Im folgenden Kapitel wird daher zunächst beschrieben, wie diese Methode zur systematischen Beschreibung von physikalischen Effekten adaptiert wird. Darauf aufbauend werden die Zusammenhänge zu physikalischen Gesetzmäßigkeiten und die systematische Aktionsbeschreibung in Form von Aktionsprimitiven, beziehungsweise Aktionsprimitivnetzen beschrieben.

Name	Symbol	Beschreibung
<i>Wandeln</i>		Eine Energie oder Energiekomponente A in eine Energie oder Energiekomponente B wandeln.
<i>Vergrößern/ Verkleinern</i>		Den skalaren Wert a_1 einer Energiekomponente auf einen Wert a_2 vergrößern oder verkleinern.
<i>Richtung ändern</i>		Die Richtung einer vektoriellen Energiekomponente ändern.
<i>Leiten</i>		Einen „bestimmten Weg bereiten“, um eine Energie oder Energiekomponente von Ort 1 nach Ort 2 zu bringen.
<i>Isolieren</i>		Verhindern, dass eine Energie oder Energiekomponente in einen bestimmten Raum hinein wirken kann.

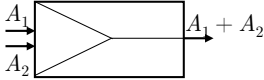
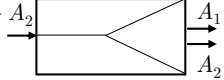
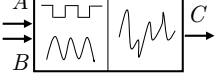
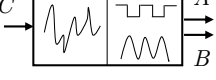
<i>Sammeln</i>		Mehrere Mengen gleicher oder verschiedener Energien oder Energiekomponenten zusammenfassen (addieren, summieren, bündeln).
<i>Teilen</i>		Eine Energie oder Energiekomponente (Kraft, Bewegung) in mehrere Energiemengen oder Energiekomponenten teilen.
<i>Mischen</i>		Energien verschiedener Qualität zusammenbringen (mischen).
<i>Trennen</i>		Energien nach Unterscheidungsmerkmalen ordnen (sortieren).

Tabelle 3.1: Liste der elementaren Operationen auf Energien, beziehungsweise Energiekomponenten nach [Koller98].

3.2 Prinzipielle Physikalische Effekte

In folgenden Kapitel wird das Konzept der *Prinzipiellen Physikalischen Effekte* (PPEs) beschrieben. Dies umfasst die Definition der verfügbaren physikalischen Operationen sowie der als Ein- und Ausgabeparameter verwendeten physikalischen Quantitäten. Darauf aufbauend werden die Zusammenhänge zwischen einem Prinzipiellen Physikalischen Effekt, einer physikalischen Gesetzmäßigkeit und einem dazugehörigen Aktionsprimitivnetz erläutert.

Physikalische Größen

Zunächst werden die verwendeten physikalischen Größen kategorisiert und beschrieben. Im Allgemeinen wird zwischen *Basisgrößen* und *abgeleiteten Größen* unterschieden, wobei Basisgrößen anhand der sogenannten *SI-Einheiten* (Système international d'unités) beschrieben werden und abgeleitete Größen als Potenzprodukte von Basisgrößen dargestellt werden können [ISO09]. Insgesamt gibt es sieben Basisgrößen, welche in Tabelle 3.2 dargestellt sind.

Name	Dimension	SI-Einheit
Länge	L	Meter (m)
Masse	M	Kilogramm (kg)
Zeit	T	Sekunde (s)
Stromstärke	I	Ampere (A)
Temperatur	Θ	Kelvin (K)
Stoffmenge	N	Mol (mol)
Lichtstärke	J	Candela (cd)

Tabelle 3.2: Liste der physikalischen Basisgrößen nach [ISO09].

Betrachtet man die Ausführung einer Aktion aus dem Bereich der Objektmanipulation, so lässt sich diese wie bereits beschrieben in verschiedene Phasen untergliedern. Bei der Ausführung der einzelnen Phasen handelt es sich typischerweise um mechanische Vorgänge, weshalb im Folgenden die im Rahmen von mechanischen Vorgängen manipulierten physikalischen Größen untersucht und beschrieben werden.

Die relevanten Basisgrößen für mechanische Vorgänge sind die *Länge*, *Masse* und *Zeit*. Aus diesen werden eine Reihe an weiteren Größen abgeleitet, welche typischerweise in *geometrische*, *kinematische* und *dynamische Größen* unterteilt werden [Awrejcewicz12]. Die geometrischen Größen entsprechen Potenzen der Basisgröße Länge, die kinematischen Größen entsprechen einem Potenzprodukt der Basisgrößen Zeit und Länge und die dynamischen Größen entsprechen einem Potenzprodukt der Basisgrößen Masse, Zeit und Länge. Die für mechanische Vorgänge relevanten physikalischen Größen sind in Tabelle 3.3 dargestellt.

Name	Beispiel	Dimension	Einheit
Winkel	Drehwinkel, Raumwinkel	1	Radian (rad)
Länge	Länge, Breite, Höhe, Radius, Weglänge	L	Meter (m)
Fläche	Flächeninhalt, Oberfläche	L^2	Quadratmeter (m^2)
Volumen	Volumen, Rauminhalt	L^3	Kubikmeter (m^3)
Zeit	Zeit, Dauer, Periodendauer	T	Sekunde (s)
Frequenz	Frequenz, Drehzahl	T^{-1}	Hertz (Hz)
Winkelgeschwindigkeit	Winkelgeschwindigkeit, Kreisfrequenz	T^{-1}	$rad \cdot s^{-1}$
Winkelbeschleunigung	Winkelbeschleunigung	T^{-2}	$rad \cdot s^{-2}$
Geschwindigkeit	Geschwindigkeit	LT^{-1}	$m \cdot s^{-1}$
Beschleunigung	Beschleunigung, Fallbeschleunigung	LT^{-2}	$m \cdot s^{-2}$
Ruck	Ruck	LT^{-3}	$m \cdot s^{-3}$
Masse	Masse, Gewicht	M	Kilogramm (kg)
Kraft	Kraft, Gewichtskraft, Reibungskraft	MLT^{-2}	Newton (N)
Impuls	Impuls, Kraftstoß	MLT^{-1}	N·s
Drehmoment	Drehmoment, Torsionsmoment	ML^2T^{-2}	N·m
Drehimpuls	Drehimpuls, Drall, Impulsmoment	ML^2T^{-1}	N·m·s
Energie	Arbeit, Energie	ML^2T^{-2}	Joule (J)
Leistung	Leistung	ML^2T^{-3}	Watt (W)
Dichte	Dichte, Massendichte	ML^{-3}	$kg \cdot m^{-3}$
Druck	Druck, Luftdruck, Spannung	$ML^{-1}T^{-2}$	Pascal (Pa)

Tabelle 3.3: Liste der relevanten geometrischen (graue Hervorhebung), kinematischen (grau-blaue Hervorhebung) und dynamischen (blaue Hervorhebung) Größen, welche im Rahmen von mechanischen Vorgängen verwendet werden (nach [Awrejcewicz12]).

Physikalische Operationen

Betrachtet man die elementaren Operationen nach [Koller98] (siehe Tabelle 3.1) so lassen sich diese bei Verwendung von allgemeinen physikalischen Größen teilweise zusammenfassen. Die zulässigen Zusammenfassungen werden im Folgenden erläutert.

Beispielsweise lassen sich die Operationen *Vergrößern/Verkleinern*, *Richtung ändern* und *Leiten* zu einer physikalischen Operation *change* zusammenfassen, welche den Wert und/oder die Richtung einer physikalischen Größe ändert. Bei skalaren Größen (z.B. Masse, Zeit) entspricht diese Operation einer Wertänderung, bei vektoriellen Größen (z.B. Kraft, Geschwindigkeit) kann dies sowohl einer Wert- als auch einer Richtungsänderung entsprechen. Die in [Koller98] vorgenommene Unterteilung zwischen skalaren und vektoriellen Operationen ist somit nicht notwendig und kann durch eine abstraktere Operation zusammengefasst werden, welche je nach Eingangsgröße eine skalare, vektorielle oder kombinierte Operation beschreibt.

Weitere Zusammenfassungen sind für die elementaren Operationen *Sammeln* und *Mischen*, sowie für *Teilen* und *Trennen* möglich. Die erste Kombination wird zu einer physikalischen Operation *merge* zusammengefasst, welche eine Menge physikalischer Größen entgegennimmt und zu einer neuen physikalischen Größe kombiniert. Die dazu inverse Funktion wird als die physikalische Operation *split* umschrieben. Diese nimmt eine physikalische Größe entgegen und teilt diese in eine Menge von physikalischen Größen. Auch an dieser Stelle wird die in [Koller98] vorgenommene Unterteilung durch jeweils eine abstraktere Operation zusammengefasst, welche die unterschiedlichen Ausprägungen beinhaltet.

Die elementaren Operationen *Wandeln* und *Isolieren* werden sinngemäß in Form der physikalischen Operationen *transform* und *absorb* übernommen. Daher lassen sich die fünf physikalischen Operationen ableiten und definieren, welche in Tabelle 3.4 dargestellt sind.

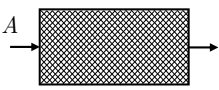
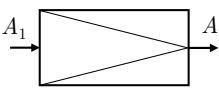
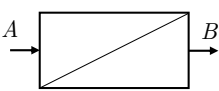
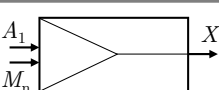
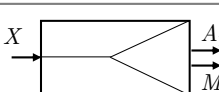
Name	Symbol	Beschreibung
<i>absorb</i>		Absorbiere eine physikalische Größe A
<i>change</i>		Ändere den Wert und/oder die Richtung einer physikalischen Größe von A_1 in A_2
<i>transform</i>		Transformiere eine physikalische Größe vom Typ A in eine physikalische Größe vom Typ B
<i>merge</i>		Verbinde eine Menge von physikalischen Größen $A_1 \dots M_n$ zu einer physikalischen Größe X
<i>split</i>		Teile eine physikalische Größe X in die physikalischen Größen $A_1 \dots M_n$

Tabelle 3.4: Liste der *Physikalischen Operationen*. Diese werden unterteilt in *elementare* Operationen (*absorb*, *change*, *transform*) und *komplexe* Operationen (*merge*, *split*).

Da nun sowohl die physikalischen Größen als auch die grundsätzlichen physikalischen Operationen festgelegt sind, kann die Menge der *Prinzipiellen Physikalischen Effekte* definiert werden.

Sei $\mathcal{Q} := \{angle, length, \dots, pressure\}$ die Menge der *physikalischen Größen* (siehe Tabelle 3.3).

Sei $\mathcal{U} := \{m, kg, s, \dots\}$ die Menge der *physikalischen Einheiten*.

Sei $\mathcal{V} := \{\mathbb{N}, \mathbb{R}, \mathbb{R}^2, \dots\}$ die Menge der *Wertebereiche*.

Sei $\widetilde{\mathcal{PQ}} := \{(v', u', q') \mid v' \in \mathcal{V}, u' \in \mathcal{U}, q' \in \mathcal{Q}\}$ die Menge der *physikalischen Qualitäten*.

Sei $\mathcal{PQ} := \{(v, u, q) \mid v \in \mathcal{V}, u = u', q = q'\}$ die Menge der zu einer physikalischen Qualität $pq \in \widetilde{\mathcal{PQ}}$ dazugehörigen *physikalischen Quantitäten*.

Sei $\mathcal{PO} := \{absorb, change, transform, merge, split\}$ die Menge der *physikalischen Operationen*.

Sei α die Anzahl der *Eingabeparameter* für eine physikalische Operation $o \in \mathcal{PO}$. Daraus folgt:

$$\alpha = \begin{cases} 1 & o \in \{absorb, change, transform, split\} \\ a & o \in \{merge\}, a \in 2, \dots, k, k \in \mathbb{N} \end{cases}$$

Sei β die Anzahl der *Ausgabeparameter* für eine physikalische Operation $o \in \mathcal{PO}$. Daraus folgt:

$$\beta = \begin{cases} 0 & o \in \{absorb\} \\ 1 & o \in \{change, transform, merge\} \\ b & o \in \{split\}, b \in 2, \dots, l, l \in \mathbb{N} \end{cases}$$

Dann lässt sich die Menge der *Prinzipiellen Physikalischen Effekte* \mathcal{PPE} wie folgt definieren:

$$\mathcal{PPE} := \{(T, I, O) \mid T \in \mathcal{PO}, I \in \mathcal{PQ}^\alpha, O \in \mathcal{PQ}^\beta\} \quad (3.1)$$

Die Prinzipiellen Physikalischen Effekte werden in *elementare*, *komplexe* und *zusammengesetzte* Effekte kategorisiert. Ein Prinzipieller Physikalischen Effekt ist elementar, falls genau ein Eingabeparameter und kein oder genau ein Ausgabeparameter vorhanden ist. Ein Prinzipieller Physikalischen Effekt ist komplex, falls mehr als ein Eingabe- oder Ausgabeparameter vorhanden ist. Ein Prinzipieller Physikalischen Effekt ist zusammengesetzt, falls dieser durch eine Menge elementarer und komplexer physikalischer Effekte beschrieben werden kann.

Im Rahmen dieser Arbeit werden zunächst ausschließlich die elementaren physikalischen Effekte *absorb*, *change* und *transform* untersucht. Die komplexen und zusammengesetzten Effekte werden im Rahmen von nachfolgenden Arbeiten untersucht (siehe Kapitel 7.2).

Zusammenhänge zwischen PPEs und physikalischen Gesetzmäßigkeiten

Der nächste Schritt besteht in der Zuordnung einer physikalischen Gesetzmäßigkeit zu einem spezifischen PPE, welche typischerweise bei der Ausführung benötigt wird. Hierbei kann die physikalische Gesetzmäßigkeit beispielsweise dazu genutzt werden, zur Ausführung benötigte, nicht explizit spezifizierte Parameter festzulegen.

Zunächst werden die allgemeinen Zusammenhänge beschrieben, die konkrete Nutzung und Umsetzung erfolgt bei der Definition des zu einem PPE zugeordneten Aktionsprimitivnetz MPN, was im nächsten Kapitel beschrieben wird.

Eine physikalische Gesetzmäßigkeit lässt sich im Allgemeinen darstellen als eine Funktion

$$A = f(q_1, \dots, q_n) \quad A \in \mathcal{PQ}_A \quad q_1 \in \mathcal{PQ}_1, \dots, q_n \in \mathcal{PQ}_n \quad n \in \mathbb{N} \quad (3.2)$$

beziehungsweise als eine Abbildung der Form:

$$\mathcal{PL} : \mathcal{PQ}_1 \times \dots \times \mathcal{PQ}_n \rightarrow \mathcal{PQ}_A \quad (3.3)$$

Neben einer solchen physikalischen Gesetzmäßigkeit kann die Ausführung eines PPE zudem eine Reihe an Nebenbedingungen erfordern. Ist beispielsweise zur erfolgreichen Ausführung ein kontinuierlicher Kontakt zwischen einem Objekt und der Umgebung notwendig, so kann dies anhand einer entsprechenden Nebenbedingung spezifiziert werden. Hierbei wird zwischen *erforderlichen* und *optionalen Nebenbedingungen* unterschieden. Die erforderlichen Nebenbedingungen werden zur erfolgreichen Ausführung eines spezifischen PPE benötigt. Die optionalen Nebenbedingungen können beispielsweise Sicherheitskriterien wie die Limitierung einer Maximalkraft beschreiben. Eine Nebenbedingung lässt sich in allgemeiner Form als Relation auf physikalische Quantitäten wie folgt beschreiben

$$\mathcal{R} \subseteq \mathcal{PQ}_1 \times \dots \times \mathcal{PQ}_n \quad (3.4)$$

mit

$$\mathcal{R} = \{(pq_i, \dots, pq_n) \in \mathcal{PQ}_1 \times \dots \times \mathcal{PQ}_n \mid z(pq_i, \dots, pq_n)\} \quad (3.5)$$

wobei $z(pq_i, \dots, pq_n)$ eine Prädikatsfunktion definiert, welche die Relation zwischen den physikalischen Quantitäten beschreibt. Diese entspricht einer Abbildung der Form:

$$z : \mathcal{PQ}_1 \times \dots \times \mathcal{PQ}_n \rightarrow \{true, false\} \quad (3.6)$$

Die allgemeinen Zusammenhänge zwischen den betrachteten PPE, sowie der zugeordneten physikalischen Gesetzmäßigkeit \mathcal{PL} und den erforderlichen physikalischen Nebenbedingungen \mathcal{R}_{req} sind in Tabelle 3.5 dargestellt. Im Folgenden werden die Eigenschaften der unterschiedlichen Typen von PPE beschrieben.

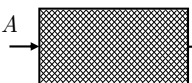
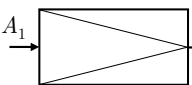
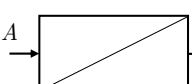
Name	Symbol	Phy. Gesetz	Nebenbedingungen
<i>absorb</i>		$A = f(q_1, \dots, q_n)$	erforderliche NB, z.B. $A < A_{\text{ref}}, A > A_{\text{ref}}$
<i>change</i>		$A_1 = f(q_{11}, \dots, q_{1n})$ $A_2 = f(q_{21}, \dots, q_{2n})$	erforderliche NB, z.B. $A_1 > A_2, A_{1,x} < A_{2,x}$
<i>transform</i>		$A = f_1(q_1, \dots, q_n, B)$ $B = f_2(q_1, \dots, q_n, A)$	erforderliche NB $q_1 = c_1, \dots, q_n = c_n$

Tabelle 3.5: Zusammenhänge zwischen den elementaren physikalischen Effekten (*absorb*, *change*, *transform*), der zugeordneten physikalischen Gesetzmäßigkeit und während der Effektausführung relevanten physikalischen Nebenbedingungen.

Zusammenhänge bei prinzipiellen physikalischen Effekten vom Typ *absorb*

Einem spezifischen PPE vom Typ *absorb* kann ein physikalisches Gesetz gemäß Gleichung 3.2, beziehungsweise Gleichung 3.3 zugeordnet werden. Da ein System typischerweise nicht in der Lage ist, eine unendliche Menge einer physikalischen Größe zu absorbieren, wird eine entsprechende Referenzgröße A_{ref} vom Typ der Eingabegröße A festgelegt. Diese darf während der Ausführung des Effekts nicht überschritten werden. Hierbei handelt es sich somit um eine erforderliche Nebenbedingung (siehe Tabelle 3.5), welche im Allgemeinen gemäß Gleichung 3.5 beschrieben werden kann. Soll beispielsweise eine Kraft F auf ein Objekt aufgebracht werden, welche einen Grenzwert F_{ref} nicht überschreiten darf, da sonst ein Schaden am Objekt entstehen würde, so kann dies durch den Effekt

$$PPE = (\text{absorb}, ((5, 0, 0), N, \text{force}), \emptyset)$$

dem physikalischen Gesetz

$$F = m \cdot a$$

$$\implies \mathcal{PQ}_1 = \{(m, \text{kg}, \text{mass})\}$$

$$\mathcal{PQ}_2 = \{(a, m \cdot s^{-2}, \text{acc})\}$$

$$\mathcal{PQ}_A = \{(F, N, \text{force})\}$$

$$\mathcal{PL} : \mathcal{PQ}_1 \times \mathcal{PQ}_2 \rightarrow \mathcal{PQ}_A$$

$$(pq_1, pq_2) \mapsto \{(F, N, \text{force}) \mid F = m \cdot a\}$$

und der erforderlichen Nebenbedingung

$$F < F_{\text{ref}}$$

$$\implies \mathcal{PQ}_3 = \{(F, N, \text{force}) \mid F \in \mathbb{R}^3\}$$

$$\mathcal{PQ}_4 = \{(c, N, \text{force}) \mid c = F_{\text{ref}}\}$$

$$\mathcal{R} = \{(pq_3, pq_4) \in \mathcal{PQ}_3 \times \mathcal{PQ}_4 \mid z(pq_3, pq_4)\}$$

$$z : (pq_3, pq_4) \mapsto \begin{cases} true & \text{wenn } F < F_{ref} \\ false & \text{sonst} \end{cases}$$

beschrieben werden. Bei der Ausführung des Effekts kann der Wert von F_{ref} beispielsweise der im Rahmen der Instantiierung verwendeten $(5, 0, 0)$ entsprechen.

Zusammenhänge bei prinzipiellen physikalischen Effekten vom Typ *change*

Den PPE vom Typ *change* kann ebenfalls ein physikalisches Gesetz gemäß Gleichung 3.2, beziehungsweise Gleichung 3.3 zugeordnet werden. Dieses wird jedoch einmal mit dem Parametersatz q_{11}, \dots, q_{1n} und einmal mit dem Parametersatz q_{21}, \dots, q_{2n} instantiiert. Da im Rahmen dieses Effekts der Wert und/oder die Richtung einer physikalischen Größe geändert werden soll, wird die erwartete Änderung in Form einer erforderlichen Nebenbedingung formuliert. Soll beispielsweise die potentielle Energie eines Objekts durch eine Änderung der Position erreicht werden, so kann dies durch den Effekt

$$PPE = (change, (1, J, energy), (2, J, energy))$$

die physikalischen Gesetzmäßigkeiten

$$W_{pot,1} = m \cdot g \cdot h_1$$

$$W_{pot,2} = m \cdot g \cdot h_2$$

$$\Rightarrow \mathcal{PQ}_{11} = \mathcal{PQ}_{21} = \{(m, kg, mass)\}$$

$$\mathcal{PQ}_{12} = \mathcal{PQ}_{22} = \{(g, m \cdot s^{-2}, acc)\}$$

$$\mathcal{PQ}_{13} = \{(h_1, m, length)\}$$

$$\mathcal{PQ}_{23} = \{(h_2, m, length)\}$$

$$\mathcal{PQ}_{A,1} = \{(W_{pot,1}, J, energy)\}$$

$$\mathcal{PQ}_{A,2} = \{(W_{pot,2}, J, energy)\}$$

$$\mathcal{PL}_1 : \mathcal{PQ}_{11} \times \mathcal{PQ}_{12} \times \mathcal{PQ}_{13} \rightarrow \mathcal{PQ}_{A,1}$$

$$(pq_{11}, pq_{12}, pq_{13}) \mapsto \{(W_{pot,1}, J, energy) \mid W_{pot,1} = m \cdot g \cdot h_1\}$$

$$\mathcal{PL}_2 : \mathcal{PQ}_{21} \times \mathcal{PQ}_{22} \times \mathcal{PQ}_{23} \rightarrow \mathcal{PQ}_{A,2}$$

$$(pq_{21}, pq_{22}, pq_{23}) \mapsto \{(W_{pot,2}, J, energy) \mid W_{pot,2} = m \cdot g \cdot h_2\}$$

und die erforderliche Nebenbedingung

$$W_{pot,1} > W_{pot,2}$$

$$\begin{aligned}
 \implies \mathcal{PQ}_1 &= \{(W_{pot,1}, J, energy) \mid W_{pot,1} \in \mathbb{R}\} \\
 \mathcal{PQ}_2 &= \{(W_{pot,2}, J, energy) \mid W_{pot,2} \in \mathbb{R}\} \\
 \mathcal{R} &= \{(pq_1, pq_2) \in \mathcal{PQ}_1 \times \mathcal{PQ}_2 \mid z(pq_1, pq_2)\} \\
 z : (pq_1, pq_2) &\mapsto \begin{cases} true & \text{wenn } W_{pot,1} > W_{pot,2} \\ false & \text{sonst} \end{cases}
 \end{aligned}$$

beschrieben werden. Werden wie im Beispiel die definierten Werte für $W_{pot,1}$, beziehungsweise $W_{pot,2}$ verwendet, so liegt an dieser Stelle eine fehlerhafte Parametrisierung vor, da unter der gewählten Nebenbedingung $W_{pot,1} > W_{pot,2}$ und der daraus resultierenden Prädikatfunktion kein gültiges Tupel zurück geliefert wird.

Zusammenhänge bei prinzipiellen physikalischen Effekten vom Typ *transform*

Einem spezifischen PPE vom Typ *transform* werden zwei physikalische Gesetzmäßigkeiten gemäß Gleichung 3.2, beziehungsweise Gleichung 3.3 zugeordnet, welche zum einen die Berechnung der Größe vom Eingangstyp A und zum anderen die Größe vom Ausgangstyp B beschreiben. An dieser Stelle wird gefordert, dass lediglich die Komponente B im Rahmen der ersten, beziehungsweise Komponente A im Rahmen der zweiten physikalischen Gesetzmäßigkeit als Variable und somit während der Ausführung des Effekts verändert werden kann. Ist dies nicht der Fall, so ist der entsprechende PPE vom Typ *split* anstatt vom Typ *transform*.

Betrachtet man auch hier einen exemplarischen Effekt

$$PPE = (transform, (A, Nm, torque), (B, m, length))$$

so können diesem beispielsweise die physikalische Gesetzmäßigkeiten

$$\begin{aligned}
 M &= F \cdot l \\
 l &= F^{-1} \cdot M
 \end{aligned}$$

zugeordnet werden. Die Definition der Abbildung erfolgt analog mit

$$\begin{aligned}
 \implies \mathcal{PQ}_A &= \{(M, Nm, torque)\} \\
 \mathcal{PQ}_B &= \{(l, m, length)\} \\
 \mathcal{PQ}_1 &= \{(F, N, force)\} \\
 \mathcal{PL}_1 : \mathcal{PQ}_1 \times \mathcal{PQ}_B &\rightarrow \mathcal{PQ}_A \\
 (pq_1, pq_B) &\mapsto \{(M, Nm, torque) \mid M = F \cdot l\} \\
 \mathcal{PL}_2 : \mathcal{PQ}_1 \times \mathcal{PQ}_A &\rightarrow \mathcal{PQ}_B \\
 (pq_1, pq_A) &\mapsto \{(l, m, length) \mid l = F^{-1} \cdot M\}
 \end{aligned}$$

Abschließend lässt sich festhalten, dass zur Ausführung eines PPE vom Typ *absorb*, *change* oder *transform* eine oder mehrere physikalische Gesetzmäßigkeiten sowie eine Reihe von physikalischen Nebenbedingungen festgelegt werden müssen. Die physikalischen Gesetzmäßigkeiten werden typischerweise zur Bestimmung von physikalischen Größen genutzt, welche nicht explizit genannt, jedoch zur Ausführung einer Aktion benötigt werden. Die physikalischen Nebenbedingungen beschreiben die Bedingungen, welche während der Ausführung einer Aktion eingehalten werden müssen. Betrachtet man die Verwendung im Rahmen dieser Arbeit, so wird innerhalb einer symbolischen Aktionsanweisung typischerweise keiner der benötigten Parameter oder einzuhaltenden Nebenbedingungen explizit formuliert, sondern durch die im Rahmen der Instruktion verwendeten Symbole implizit festgelegt, weshalb diese Information zur Ausführung eines entsprechenden Effekts durch das Robotersystem ermittelt werden muss.

Die beschriebenen Zusammenhänge und Informationen werden bei der Definition des zu einem PPE zugeordneten Aktionsprimitivnetzes MPN verwendet. Daher werden die physikalischen Gesetzmäßigkeiten und Nebenbedingungen nicht explizit verwendet, sondern implizit in der Definition des MPN berücksichtigt. Die Festlegung eines entsprechenden MPN zu einem PPE wird im folgenden Kapitel beschrieben.

Zusammenhänge zwischen PPEs und Aktionsprimitivnetzen

Das zu einem PPE zugeordnete Aktionsprimitivnetz MPN stellt die wesentliche Komponente eines PPE dar, da darin die während der Ausführung benötigten physikalischen Größen und Nebenbedingungen in einer parametrisierbaren Form hinterlegt sind. Somit beschreibt das zugeordnete MPN die Semantik eines auszuführenden PPE.

Wie im letzten Kapitel dargestellt, kann ein PPE eine Menge an physikalischen Nebenbedingungen erfordern, welche die Ausführung des Effekts beeinflussen und somit Varianten einer physikalischen Operation auf physikalische Quantitäten beschreiben. An dieser Stelle sei auf die allgemeinen Zusammenhänge zwischen physikalischen Effekten und Nebenbedingungen verwiesen (siehe Tabelle 3.5). Betrachtet man exemplarisch die Effekte vom Typ *change*

$(change, A_1, A_2)$ mit der impliziten Nebenbedingung $A_1 < A_2$ und

$(change, A_1, A_2)$ mit der impliziten Nebenbedingung $A_1 > A_2$

so werden aufgrund der Nebenbedingung verschiedene Ausführungen, beziehungsweise eine unterschiedliche Semantik beschrieben.

Da dies mit der bisherigen Definition der Menge \mathcal{PPE} (siehe Gleichung 3.1) nicht dargestellt werden kann und der entsprechende Zusammenhang in Form eines MPN formuliert wird, folgt die um ein entsprechendes MPN erweiterte Definition der Menge der *Prinzipiellen Physikalischen Effekte* \mathcal{PPE} mit

$$\mathcal{PPE} := \{ (T, I, O, MPN) \mid T \in \mathcal{PO}, I \in \mathcal{PQ}^\alpha, O \in \mathcal{PQ}^\beta \} \quad (3.7)$$

Wie bereits in Kapitel 2.2 beschrieben und anhand der formalen Definition ersichtlich, beschreibt ein Aktionsprimitivnetz MPN (siehe Gleichung 2.9) einen gerichteten Graphen, welcher Aktionsprimitive \mathcal{MP} (siehe Gleichung 2.1) als Knoten und Abbruchbedingungen von Aktionsprimitiven als Kanten enthält. Daher stellt sich zunächst die Frage, welche allgemeine Struktur ein MPN im Kontext der Beschreibung von PPEs annehmen kann.

Dies umfasst zum einen die Frage nach der generellen Anzahl der Knoten, sowie die Festlegung der möglichen Startknoten MP_{start} , beziehungsweise Endknoten MP_{stop} . Zum anderen umfasst dies die Frage nach der Anzahl der Kanten, beziehungsweise den Bedingungen, welche zur Definition einer Kante zwischen zwei Aktionsprimitiven erfüllt sein müssen.

Neben der Struktur des MPN muss zudem untersucht werden, in welcher Form die Parameter der einzelnen \mathcal{MP} s festgelegt werden. Dies betrifft die Definition des lokalen Koordinatensystems (*task frame*), der Sollwerte und Regelstrategien für die entsprechenden Freiheitsgrade, der aktiven Werkzeugaktionen sowie der Abbruchkriterien.

Struktur der im System hinterlegten Aktionsprimitivnetze

Zur Analyse der Struktur der hinterlegten Aktionsprimitivnetze wird zunächst eine Aufgabe aus dem Bereich der Objektmanipulation untersucht. Ein Beispiel für eine Aufgabe aus dem Bereich der Objektmanipulation ist in Abbildung 3.1 dargestellt.

Bei der dargestellten Aufgabe umfasst die Annäherungsphase eine Annäherungsbewegung sowie das Aufgreifen des zu manipulierenden Objekts. Darauf anschließend folgt die Manipulationsphase, welche im Beispiel eine Positionsänderung des manipulierten Objekts beschreibt. Abschließend folgt die Abschlussphase, in der beispielsweise der Greifer geöffnet und zurück zur Startposition verfahren wird.

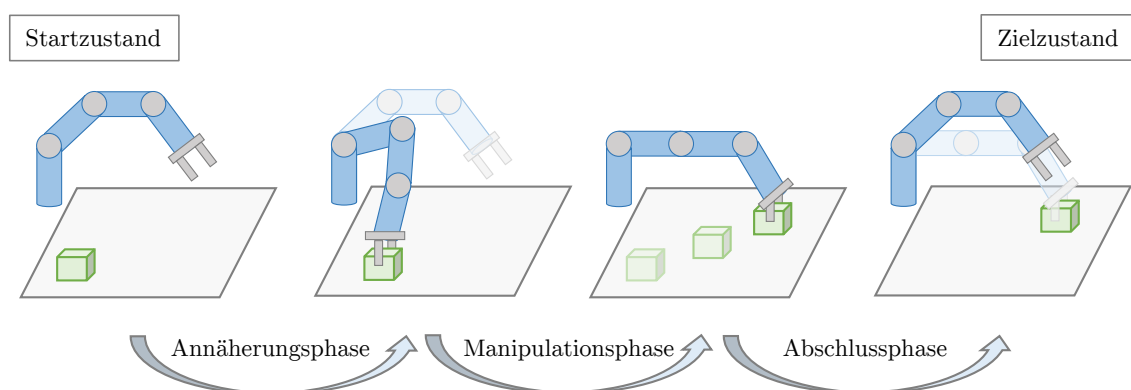


Abbildung 3.1: Beispiel für eine Aufgabe aus dem Bereich der Objektmanipulation inklusive der Unterteilung der Ausführungsphasen in die Annäherungs-, Manipulations- und Abschlussphase.

Soll die Struktur von Aktionsprimitivnetzen anhand der verschiedenen Ausführungsphasen beschrieben werden, ist zu untersuchen ob diese immer in einer festgelegten Reihenfolge auftreten. Insgesamt können die Phasen in $3! = 6$ möglichen Varianten auftreten. Diese lauten wie folgt:

V1 Greifen - Bewegen - Ablegen

V2 Greifen - Ablegen - Bewegen

V3 Bewegen - Greifen - Ablegen

V4 Bewegen - Ablegen - Greifen

V5 Ablegen - Bewegen - Greifen

V6 Ablegen - Greifen - Bewegen

Betrachtet man die Ausführung einer entsprechenden Aufgabe unter Nutzung eines Robotersystems wie in Abbildung 3.1 dargestellt (stationärer Roboterarm inklusive Greifvorrichtung), so sind ausschließlich die Varianten **V1**, **V4** und **V6** erfolgreich ausführbar und somit als gültige Aufgaben aus dem Bereich der Objektmanipulation zu klassifizieren. Betrachtet man die Varianten **V2** und **V5**, so entspricht dies keiner Aufgabe aus dem Bereich der Objektmanipulation, da die Ausführung der Bewegungsphase ohne das zu manipulierende Objekt beschrieben und somit keine Objekteigenschaft manipuliert wird. Die Variante **V3** beschreibt eine Aufgabe, welche den Zugriff auf einen bereits belegten Greifer und somit zunächst ein Ablegen des aktuell gegriffenen Objekts erfordert.

Neben der Reihenfolge muss untersucht werden, ob die entsprechenden Phasen in allen auszuführenden Aufgaben vorhanden sind. Betrachtet man eine Umgebung wie in Abbildung 3.1 dargestellt und eine exemplarische Anweisung wie *Greife den grünen Würfel!* so beinhaltet die Ausführung dieser Anweisung ausschließlich die Annäherungsphase. Um diese Gruppe von Aktionen dennoch einheitlich im Rahmen der Prinzipiellen Physikalischen Effekte beschreiben zu können, wird für jede Ausführungsphase mindestens ein Aktionsprimitiv beschrieben. Wird eine entsprechende Phase im Rahmen der Aktionsausführung nicht benötigt, so wird diese in Form eines Aktionsprimitivs ohne resultierende Roboterbewegung modelliert.

Daher werden die verschiedenen Ausführungsphasen einer Aufgabe aus dem Bereich der Objektmanipulation als Basis für die systematische Beschreibung der zugeordneten Aktionsprimitivnetze verwendet. Die allgemeine Struktur des zu einem PPE zugeordneten MPN besteht somit aus drei Komponenten, welche die Annäherungsphase, die Manipulationsphase und die Abschlussphase beschreiben und sequenziell ausgeführt werden (siehe Abbildung 3.2). Zur sequenziellen Ausführung müssen entsprechende Abbruchbedingungen zwischen den einzelnen Komponenten formuliert werden, welche den Übergang der Ausführungsphasen beschreiben.

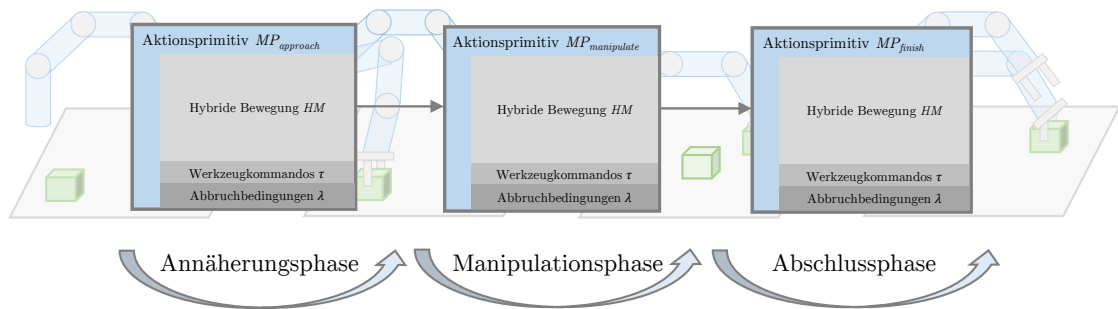


Abbildung 3.2: Allgemeine Struktur des zu einem Prinzipiellen Physikalischen Effekt zugeordneten Aktionsprimitivnetz.

Im einfachsten Fall kann eine Ausführungsphase durch genau ein Aktionsprimitiv beschrieben werden. Im Allgemeinen werden die einzelnen Phasen jedoch nicht durch ein einzelnes Aktionsprimitiv, sondern durch ein Sub-Aktionsprimitivnetz beschrieben. Hierbei lassen sich zwei wesentliche Abhängigkeiten identifizieren, welche die Struktur eines Aktionsprimitivnetzes zur Beschreibung einer Ausführungsphase beeinflussen.

Zum einen ist die Beschreibung einer Ausführungsphase abhängig von der *Komplexität der Aufgabe*. Erfordert die Ausführung eine Vielzahl an hybriden Roboterbewegungen oder Werkzeugaktionen, so kann dies nicht anhand eines einzelnen Aktionsprimitivs dargestellt werden und erfordert somit die Definition eines Aktionsprimitivnetzes.

Zum anderen ist die Beschreibung einer Ausführungsphase abhängig von der im Robotersystem *verfügbaren Sensorik*. Diese lässt sich in *lokale* und *globale Sensorik* gruppieren. In diesem Kontext erlaubt ein globaler Sensor die Bestimmung einer physikalischen Größe vor Beginn der Aktionsausführung, ein lokaler Sensor erlaubt die Bestimmung einer physikalischen Größe während der Aktionsausführung.

Ein Beispiel zum Einfluss der verfügbaren Sensorik auf das zugeordnete Aktionsprimitivnetz ist in Abbildung 3.3 dargestellt. Hier soll beispielsweise eine Annäherungsbewegung hin zu einem Objekt durchgeführt werden, dessen Position zunächst nicht exakt bekannt ist. In diesem Beispiel kann die globale Sensorik aus einer Kamera bestehen, welche zu jedem Zeitpunkt die komplette Szene erfassen kann. Zur Ausführung der Aufgabe wird die Position des zu manipulierenden Objekts benötigt. Zur Erfassung dieser Position wird ein Aktionsprimitiv MP_{detect} definiert, welches auf Basis der sensorischen Information die benötigte Position ermitteln soll. Da der globale Sensor nach Definition diese zu jeder Zeit zur Verfügung stellen kann, wird dieses Aktionsprimitiv abgearbeitet und anschließend die Annäherungsbewegung ausgeführt, welche im Rahmen des exemplarischen Aktionsprimitivs $MP_{approach}$ beschrieben wird.

Im Gegensatz dazu stellt im Beispiel eine am Robotersystem montierte Kamera einen lokalen Sensor dar, da dieser einen von der aktuellen Roboterposition abhängigen Sichtbereich besitzt. Ist das gesuchte Objekt nicht im Sichtfeld der Kamera, so muss dieses zunächst anhand einer geeigneten Suchstrategie aufgefunden werden. Hierzu wird exemplarisch ein zusätzliches Aktionsprimitiv MP_{search} definiert, welches die auszuführende Suchbewegung spezifiziert. Insgesamt wird somit ein iterativer Such- und Prüfungsvorgang durchgeführt, bis das gewünschte Objekt sensorisch erfasst und die entsprechende Position extrahiert werden kann. Abschließend kann auch an dieser Stelle die entsprechende Annäherungsbewegung ausgeführt werden.

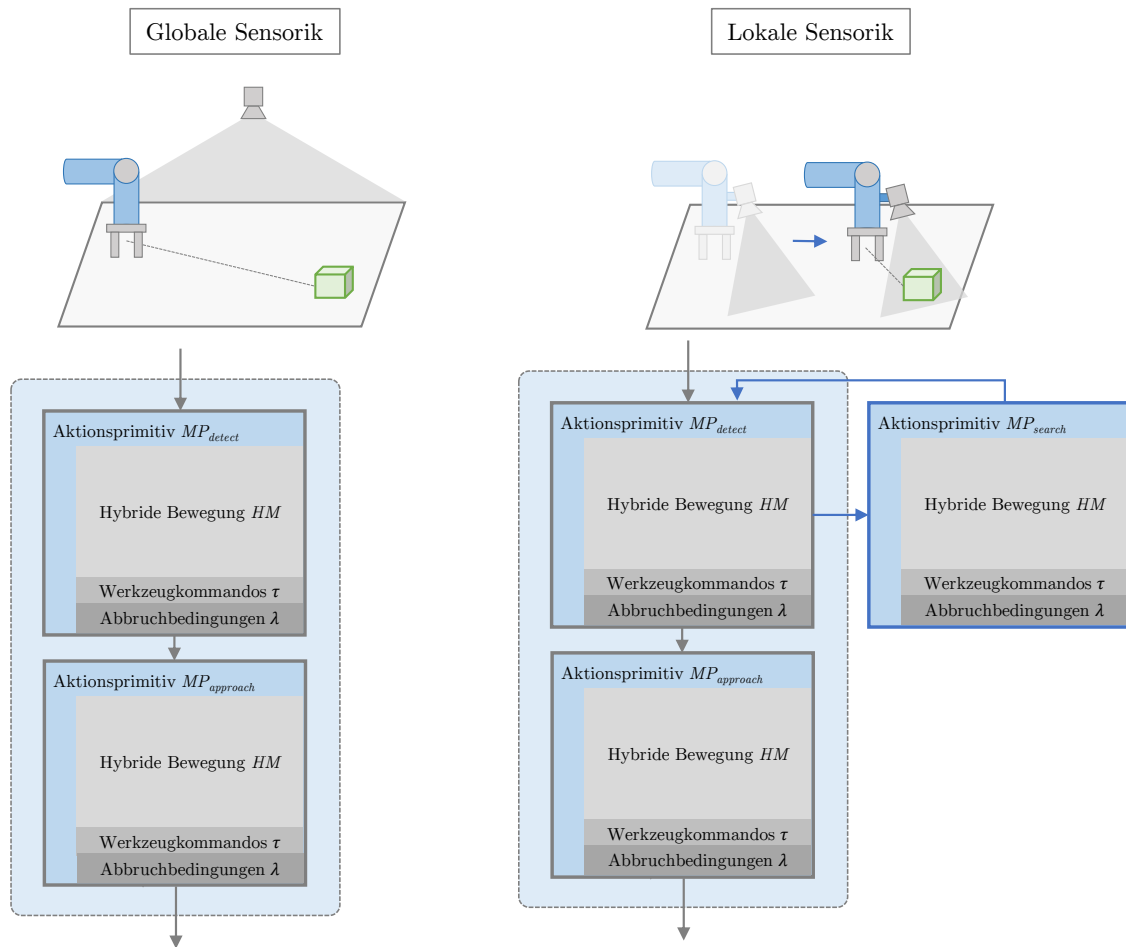


Abbildung 3.3: Beispiel für die Abhängigkeit des Aktionsprimitivnetzes von der im Robotersystem verfügbaren Sensorik. Im linken Teil der Abbildung ist eine exemplarische Szene sowie das zur Aktionsausführung benötigte Aktionsprimitivnetz für die Nutzung von globaler Sensorik dargestellt. Im rechten Teil der Abbildung ist eine exemplarische Szene sowie das zur Aktionsausführung unter Nutzung lokaler Sensorik erweiterte Aktionsprimitivnetz dargestellt.

Anhand des Beispiels wird gezeigt, dass für die Nutzung von lokaler Sensorik ein erweitertes Aktionsprimitivnetz benötigt wird. Betrachtet man die unterschiedlichen Aktionsprimitivnetze für die beiden Sensorvarianten, so sind die Aktionsprimitive für die globale Sensorik im Aktionsprimitivnetz für die lokale Sensorik enthalten. Um eine möglichst flexible Aktionsbeschreibung umzusetzen, werden daher die im Rahmen dieser Arbeit definierten Aktionen im Allgemeinen auf Basis lokaler Sensorik spezifiziert und im Robotersystem hinterlegt.

Parameterbestimmung bei Aktionsprimitiven eines hinterlegten Aktionsprimitivnetzes

Im Rahmen der Parameter der einzelnen Aktionsprimitive wird die Semantik der auszuführenden Aktion spezifiziert. Dies umfasst somit die Beschreibung der physikalischen Größen, welche während der Aktionsausführung benötigt werden, sowie der physikalischen Nebenbedingungen, welche zur erfolgreichen Ausführung einer Aktion festgelegt werden müssen.

Zunächst ist jedoch das lokale Koordinatensystem (*task frame*) \mathcal{TF} festzulegen, in welchem die auszuführende Aktion beschrieben werden soll. Zur Konstruktion des \mathcal{TF} werden eine Reihe an Möglichkeiten festgelegt, welche die Lage und Orientierung beschreiben können. Zur Beschreibung eines konkreten Effekts muss die entsprechende Konstruktion im System hinterlegt werden. Die vorgeschlagenen Möglichkeiten zur Konstruktion des \mathcal{TF} sind in Abbildung 3.4 dargestellt.

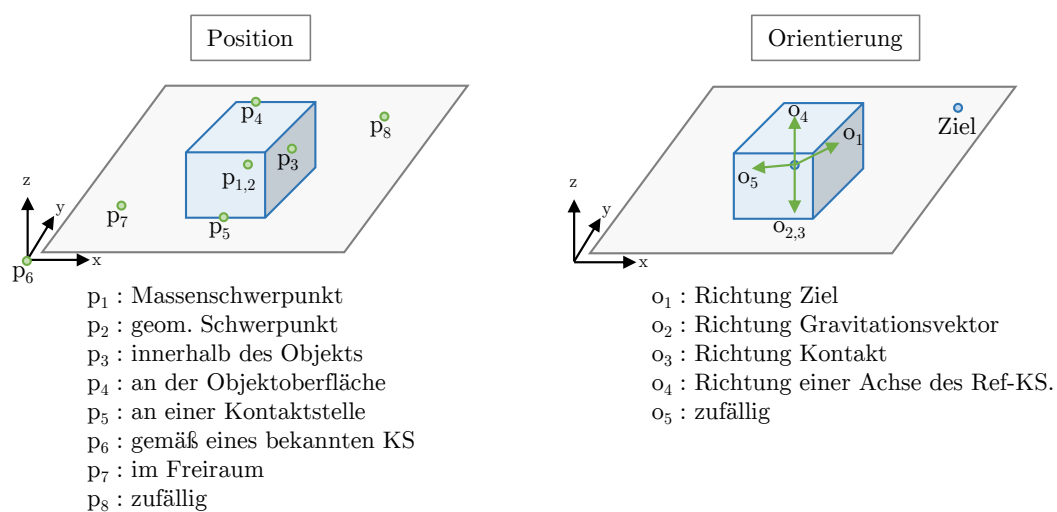


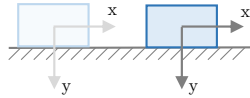
Abbildung 3.4: Möglichkeiten zur Festlegung der Position (einer Koordinate) und Orientierung (einer Achse) des lokalen Koordinatensystems (*task frame*) \mathcal{TF} einer auszuführenden Aktion.

Soll beispielsweise eine Transferbewegung im Freiraum definiert werden, bei der keine spezifischen Nebenbedingungen eingehalten werden müssen, kann der \mathcal{TF} beispielsweise dem globalen Weltkoordinatensystem entsprechen, wodurch beispielsweise die Möglichkeiten p_6 und o_4 zur Konstruktion verwendet werden. Soll beispielsweise die Ausführung eines Effekts beschrieben werden, welcher einen permanenten Kontakt zwischen manipulierten Objekt und Umgebung erfordert, so sind beispielsweise die Möglichkeiten p_5 und o_3 zur Konstruktion des benötigten \mathcal{TF} relevant.

An dieser Stelle lässt sich jedoch ein Nachteil der verwendeten lokalen Koordinatensysteme erkennen. Da alle Nebenbedingungen gemäß des lokalen Koordinatensystems beschrieben werden, kann es bei spezifischen Situationen zu Problemen bei der Ausführung kommen, sobald mehr als ein Kontaktpunkt vorhanden ist. Exemplarisch wird eine Situation erläutert (siehe Abbildung 3.5) in der ein Fläche-Fläche Kontakt zwischen Werkstück und Umgebung vorhanden ist, welcher im Rahmen des lokalen Koordinatensystems durch einen Kontaktpunkt approximiert wird. Analoge Beispiele lassen sich für weitere Kontaktsituationen konstruieren.

Situation 1:

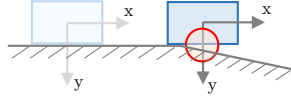
- Orientierung des TF fixiert
- Fläche-Fläche Kontakt
- Ebene Umgebung



→ Ausführung möglich

Situation 2:

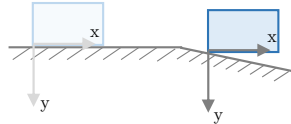
- Orientierung des TF fixiert
- Fläche-Fläche Kontakt
- unebene Umgebung



→ Nebenbedingung
nicht eingehalten

Situation 3:

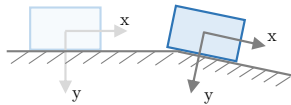
- Orientierung des TF fixiert
- Fläche-Fläche Kontakt
- unebene Umgebung



→ Ausführung möglich
→ Degenerierter Kontakt

Situation 4:

- Orientierung des TF flexibel
- Fläche-Fläche Kontakt
- unebene Umgebung



→ Ausführung möglich

Abbildung 3.5: Beispiele für Probleme bei Verwendung eines lokalen Koordinatensystems, falls bei Ausführung einer Anweisung nicht ausschließlich ein Kontaktpunkt vorhanden ist.

Im Beispiel soll eine Bewegung entlang der x -Richtung des lokalen Koordinatensystems durchgeführt werden, welche zudem die vorhandene Kontaktsituation aufrecht erhält. Daraus folgt, dass der Freiheitsgrad y mit einer entsprechenden Kontaktkraft als Sollwert kraftgeregelt ausgeführt werden muss.

In der ersten Situation wird zunächst die Position des \mathcal{TF} gemäß des Schwerpunkts des Werkstücks festgelegt. Zudem wird die Orientierung des \mathcal{TF} fixiert. Bei der dargestellten Umgebung ist die Bewegung mit einer definierten Kontaktkraft sowie des fixierten \mathcal{TF} ausführbar, das heißt zu jedem Zeitpunkt kann die definierte Kontaktkraft gemessen und der Kontakt zwischen dem festgelegten Kontaktpunkt und der Umgebung aufrecht erhalten werden.

In der zweiten Situation ist eine veränderte Umgebung dargestellt. Wie in der Abbildung dargestellt, kann dies dazu führen, dass die spezifizierte Nebenbedingung nicht zu jedem Zeitpunkt aufrecht erhalten werden kann. Im Beispiel ist kein realer Kontakt zwischen dem spezifizierten Kontaktpunkt und der Umgebung vorhanden. Da das Werkstück jedoch noch in Kontakt mit der Umgebung steht, kann ein Moment sensorisch erfasst werden, welches in Abhängigkeit zur geforderten Kontaktkraft steht. Somit wird durch das resultierende Moment eine angebliche Kontaktkraft gemessen und die Aktion weiter ausgeführt. Im Beispiel hätte dies zur Folge, dass der anfängliche Fläche-Fläche Kontakt zu einem Kante-Fläche, beziehungsweise Punkt-Fläche Kontakt degenerieren kann.

In der dritten Situation wird die Position des \mathcal{TF} verändert. Dies hat zur Folge, dass die spezifizierte Bewegung ausgeführt werden kann, jedoch der anfängliche Kontakt zu einem Kante-Fläche, beziehungsweise Punkt-Fläche Kontakt degenerieren kann. Ist der Kontakt degeneriert, bleibt er in der degenerierten Form vorhanden, das heißt der ursprüngliche Fläche-Fläche Kontakt kann während der Ausführung im Allgemeinen nicht wiederhergestellt werden.

In der vierten Situation wird eine flexible Orientierung des \mathcal{TF} spezifiziert. An dieser Stelle kann beispielsweise der rotatorische Freiheitsgrad, welcher in die Bildebene hineinzeigt, drehmomentgeregelt ausgeführt werden, um so die flexible Orientierung des \mathcal{TF} umzusetzen. Dies hat zur Folge, dass zu jedem Zeitpunkt die Kontaktkraft aufrecht erhalten werden kann. Auch an dieser Stelle kann der Kontakt jedoch kurzzeitig degenerieren, genau dann wenn ein Sprung in der Kontaktnormalen festzustellen ist. Im Unterschied zu dem Beispiel in Situation 3 kann der ursprüngliche Fläche-Fläche Kontakt im Verlauf der Ausführung wiederhergestellt werden.

Neben den Parametern zur Beschreibung des lokalen Koordinatensystems müssen die Parameter zur Definition der auszuführenden Bewegung, der aktiven Werkzeugaktionen sowie die Kriterien zum Abbruch der Bewegung festgelegt werden. Hierbei beeinflussen die einem PPE zugeordneten physikalischen Größen die Parametrisierung der adaptiven Selektionsmatrix \mathcal{D} (siehe Gleichung 2.6), welche die Sollwerte und Regelstrategien für jeden Freiheitsgrad des zur Ausführung verwendeten lokalen Koordinatensystems beschreiben.

Im Allgemeinen beschreiben die Eingabeparameter eines PPE die Regelstrategien und Sollwerte der einzelnen Freiheitsgrade des zur Bewegung verwendeten lokalen Koordinatensystems. Sind diese direkt als Regelstrategie im System vorhanden, so können diese direkt spezifiziert werden. Sind diese nicht direkt als Regelstrategie vorhanden, muss eine Größe der zum PPE zugeordneten physikalischen Gesetzmäßigkeiten eine direkt regelbare Größe darstellen.

Betrachtet man hierzu das Beispiel

(
	<i>absorb</i> ,	\leftarrow <i>Effekttyp</i>
	$(\mathbf{f}_x, N, \mathbf{force})$,	\leftarrow <i>Eingabeparameter</i>
	\emptyset ,	\leftarrow <i>Ausgabeparameter</i>
	<i>MPN</i>	\leftarrow <i>zugeordnetes Aktionsprimitivnetz</i>
)		

so enthält das Aktionsprimitiv, welches die Effektausführung beschreibt, eine adaptive Selektionsmatrix der Form

$D = ($		
	$(\mathbf{f}_x, \mathbf{force})$,	\leftarrow <i>resultierende Kraftregelung des x-Freiheitsgrads</i>
	\dots	\leftarrow <i>Regelung sonstiger Freiheitsgrade</i>
)		

das heißt einer der translatorischen Freiheitsgrade wird kraftgeregelt ausgeführt. Hierbei handelt es sich um ein Beispiel, bei dem die physikalische Größe direkt als Regelstrategie im System verfügbar ist.

Betrachtet man hingegen das Beispiel

$$\begin{pmatrix}
 \text{change}, & \leftarrow \text{Effektyp} \\
 (\mathbf{W}_1, J, \text{energy}), & \leftarrow \text{Eingabeparameter} \\
 (\mathbf{W}_2, J, \text{energy}), & \leftarrow \text{Ausgabeparameter} \\
 \text{MPN} & \leftarrow \text{zugeordnetes Aktionsprimitivnetz}
 \end{pmatrix}$$

so steht dem System zur physikalischen Größe *Energie* typischerweise keine Regelstrategie zur Verfügung. Diesem Effekt wird beispielsweise die physikalische Gesetzmäßigkeit $W = m \cdot g \cdot h$ und die Nebenbedingung $W_1 < W_2$ zugeordnet (siehe Kapitel 3.2). Bei Effekten von diesem Typ wird die zugeordnete physikalische Gesetzmäßigkeit mit zwei unterschiedlichen Parametersätzen instantiiert, zum Beispiel in der Form $W_1 = m \cdot g \cdot h_1$ und $W_2 = m \cdot g \cdot h_2$.

Somit stellt hier der Parameter h_1 , beziehungsweise h_2 die Variable dar, welche den Sollwert und die Regelstrategie für einen Freiheitsgrad bestimmt. Je nach gewählter, erforderlicher Nebenbedingung entspricht der Sollwert einem der beiden Werte, welcher durch das System berechnet werden kann. Somit enthält das Aktionsprimitiv, welches die Effektausführung beschreibt, eine adaptive Selektionsmatrix der Form

$$\begin{pmatrix}
 \dots & \\
 (\mathbf{h}, \text{position}), & \leftarrow \text{resultierende Positionsregelung des Freiheitsgrads, bestimmt} \\
 & \quad \text{aus der zugeordneten physikalischen Gesetzmäßigkeit} \\
 \dots & \leftarrow \text{Regelung sonstiger Freiheitsgrade} \\
 \end{pmatrix}$$

das heißt einer der translatorischen Freiheitsgrade wird positionsgeregelt ausgeführt. Hierbei handelt es sich um ein Beispiel, bei dem die physikalische Größe nicht direkt als Regelstrategie im System verfügbar ist und somit über die zugeordnete physikalische Gesetzmäßigkeit bestimmt werden muss.

Die Ausgangsparameter und physikalischen Nebenbedingungen eines PPE beschreiben im Allgemeinen die Abbruchkriterien des Aktionsprimitivs, welches die Effektausführung beschreibt. Diese werden typischerweise ergänzt durch standardmäßige Abbruchkriterien wie beispielsweise der Betätigung des Not-Aus oder dem Erreichen von Maximalwerten (z.B. Kräfte/Momente/Geschwindigkeiten). Betrachtet man nochmals die beiden zuvor genannten Beispiele, so werden diesen beispielsweise die Nebenbedingungen $f_x < f_{ref}$ und $W_1 < W_2$ zugeordnet. Diese können direkt als Abbruchkriterium übernommen werden.

Abschließend lässt sich festhalten, dass die Prinzipiellen Physikalischen Effekte ein Konzept beschreiben, welches es erlaubt, Roboteraktionen basierend auf Aktionsprimitivnetzen unter Nutzung von physikalischen Größen, Gesetzmäßigkeiten und Nebenbedingungen festzulegen. Im Rahmen dieses Kapitels wurden die benötigten physikalischen Größen und Operationen zunächst eingeführt und anschließend die Zusammenhänge zwischen den einzelnen Komponenten erläutert. Bisher werden ausschließlich Zusammenhänge zwischen den PPEs und der im Rahmen dieser Arbeit verwendeten, subsymbolischen Schnittstelle dargestellt. Der nächste Schritt besteht darin, die Zusammenhänge zwischen den PPEs und einer für den Menschen intuitiven, symbolischen Repräsentation darzustellen.

3.3 Verbalisierte Physikalische Effekte

In Rahmen dieses Kapitels wird das Konzept der *Verbalisierten Physikalischen Effekte* (VPE) beschrieben. Dieses stellt eine Erweiterung der im vorangegangenen Kapitel eingeführten PPEs dar. Während im Rahmen der PPEs die Zusammenhänge der Transformations- und Kontrollschicht beschrieben werden, beschreibt ein VPE die Zusammenhänge zwischen der Transformations- und Benutzerschicht.

Ein Ziel im Rahmen dieser Arbeit ist die Verarbeitung und Ausführung von symbolischen Anweisungen, bei denen typischerweise das Verb der Anweisung die auszuführende Aktion kennzeichnet. Wie im letzten Kapitel beschrieben, werden die auszuführenden Aktionen hierbei in Form von PPEs repräsentiert. Daher stellt sich zum einen die Frage, in welcher Form die Zuordnung eines Verbs zu einer Aktionsbeschreibung basierend auf PPEs umgesetzt werden kann. Zum anderen stellt sich die Frage, in welcher Form zusätzliche Angaben einer symbolischen Anweisung die Ausführung, beziehungsweise die Parametrisierung einer auszuführenden Aktion beeinflussen.

An dieser Stelle werden zunächst die möglichen Symbole und symbolischen Parameter beschrieben. Neben der Aktion in Form eines Verbs müssen weitere symbolische Parameter spezifiziert werden, um eine grammatikalisch korrekte und semantisch vollständige Anweisung zu spezifizieren. Betrachtet man beispielsweise folgende Anweisungen

“*Schiebe!*“

“*Schiebe den roten Block!*“

“*Schiebe gegen den blauen Block!*“

“*Schiebe den roten Block gegen den blauen Block!*“

so beschreiben die ersten drei Anweisungen semantisch unvollständige Instruktionen. Während bei der ersten Anweisung keinerlei Zusatzinformation spezifiziert wird, ist bei der zweiten Instruktion eine Information über das zu manipulierende Objekt spezifiziert. Unklar ist jedoch, in welche Richtung oder über welche Distanz das Objekt geschoben werden soll. Diese Information ist in der dritten Anweisung vorhanden, jedoch ist keine Information über das zu manipulierende Objekt spezifiziert. Die vierte Anweisung enthält die benötigte Information und stellt somit eine semantisch vollständige Anweisung dar.

Die Anzahl der Ergänzungen, welche einem Verb zugeordnet werden können, wird im Bereich der Linguistik als *Valenz* bezeichnet [Tesniere80, Hyvarinen03, Schumacher04]. Eine Ergänzung kann hierbei die Ausprägung *obligatorisch* oder *fakultativ* annehmen. Eine obligatorische Ergänzung wird zur semantisch vollständigen Beschreibung zwingend benötigt, eine fakultative Ergänzung entspricht einer zusätzlichen Information. Im Allgemeinen werden die Verben auf Basis der obligatorischen Ergänzungen in folgende Kategorien unterteilt [Vuillaume03]:

- *Nullwertige* Verben, zum Beispiel *(es) schneit*, *(es) blitzt*, *(es) regnet*.
- *Einwertige* Verben, zum Beispiel *atmen*, *dauern*, *leben*.
- *Zweiwertige* Verben, zum Beispiel *berühren*, *bremsen*, *warten*.
- *Dreiwertige* Verben, zum Beispiel *drehen*, *schieben*, *stoßen*.

Neben der Ausprägung wird jeder Ergänzung eine bestimmte Funktion, beziehungsweise Klasse zugeordnet. Diese wird *Ergänzungsklasse* genannt [Schumacher04]. Eine Ergänzungsklasse enthält hierbei eine syntaktische und eine semantische Dimension. Die syntaktische Dimension beschreibt, in welcher konkreten Form eine Ergänzung im Rahmen eines Satzes, beziehungsweise einer Anweisung beschrieben wird. Hierzu werden typischerweise die unterschiedlichen Wortgruppen, beziehungsweise Phrasen verwendet. Die semantische Information beschreibt, welche Form der Information im Rahmen der Ergänzung beschrieben wird. Diese wird typischerweise in Form einer Anapher dargestellt. In [Schumacher04] werden insgesamt acht Ergänzungsklassen unterschieden, welche in Tabelle 3.6 dargestellt sind.

Ergänzungsklasse	typische Ausdrucksform	Anapher
Nominativergänzung (NomE)	Nominalphrase (NP)	jemand / etwas
Akkusativergänzung (AkkE)	Nominalphrase (NP)	jemanden / etwas
Genitivergänzung (GenE)	Nominalphrase (NP)	jemandes / etwas
Dativergänzung (DatE)	Nominalphrase (NP)	jemandem / etwas
Präpositivergänzung (PräpE)	Präpositionalphrase (PP)	Präp + jemand (~) / Präp + etwas
Adverbativergänzung (AdvE)	Adverbialphrase (ADVP)	irgendwo/.../irgendwann/.../irgendwobei
Prädikativergänzung (PrädE)	Adjektivphrase (ADJP)	so sein / ein solcher/solches sein
Verbativergänzung (VerbE)	Verbalphrase (VP)	etwas (zu) tun

Tabelle 3.6: Übersicht der Ergänzungsklassen nach [Schumacher04].

Auf Basis der Bestimmung der benötigten Ergänzungen zu einem entsprechenden Verb lässt sich dadurch ein resultierendes *Satzmuster*, beziehungsweise ein resultierender *Satzbauplan* ableiten. Dieser beschreibt sowohl die obligatorischen als auch die fakultativen Ergänzungen, welche mit einer entsprechenden Verbvariante verbunden werden können.

Betrachtet man an dieser Stelle nochmals das einleitende Beispiel, so können für das darin verwendete Verb *schieben* die entsprechenden Ergänzungen, die allgemeine Struktur unter Verwendung der Anapher sowie der zugeordnete Satzbauplan angegeben werden (siehe Tabelle 3.7). Die allgemeinen Strukturen und Satzbaupläne sind im Rahmen von sogenannten Valenzwörterbüchern wie beispielsweise [IDS16] verfügbar.

Struktur:	jemand/etwas schiebt jemand/etwas irgendwohin
Satzbauplan:	NomE AkkE AdvE
Phrasen:	NP NP PP
Beispiel:	Schiebe den roten Block gegen den blauen Block!

Tabelle 3.7: Struktur, Satzbauplan und Ausdrucksformen für die Beispielanweisung “*Schiebe den roten Block gegen den blauen Block!*“. Die Struktur und der Satzbauplan sind aus [IDS16] entnommen. Die Ausdrucksform wird auf Basis eines Parsers nach [Klein03] erzeugt.

An dieser Stelle lässt sich erkennen, dass die benötigte Nominativergänzung nicht explizit spezifiziert wird. Diese stellt typischerweise das Subjekt der Anweisung dar, welches im Rahmen einer Anweisung im Imperativ implizit den Befehlsempfänger zugeordnet wird. Im Rahmen dieser Arbeit entspricht dies somit automatisch dem Robotersystem, welches die instruierte Anweisung verarbeiten und ausführen soll.

Betrachtet man nun die Parameter der im Rahmen dieser Arbeit untersuchten PPE, so besitzen diese α Eingabeparameter ($\alpha = 1$ für Effekte vom Typ *absorb*, *change* und *transform*) und β Ausgabeparameter ($\beta = 0$ für Effekte vom Typ *absorb*, $\beta = 1$ für Effekte vom Typ *change* und *transform*). Diese Parameter müssen somit im Rahmen einer der obligatorischen Ergänzungen des zu einem PPE zugeordneten Verbs in symbolischer Form beschrieben werden. Dadurch ergeben sich eine Reihe an Anforderungen an das zu einem PPE zugeordnete Verb, so muss dieses beispielsweise $\alpha + \beta + 1$ obligatorische Ergänzung aufnehmen können. Zudem muss aus einer der Ergänzungen entweder ein physikalischer Parameter *direkt* oder *indirekt* über eine physikalische Gesetzmäßigkeit extrahiert werden können. Das Auflösen der symbolischen Information wird in Kapitel 5.2 beschrieben.

Im Folgenden wird die Zuordnung eines Verbs zu einem entsprechenden PPE beschrieben. Diese wird anhand empirischer Daten vorgenommen, welche im Rahmen einer Benutzerstudie erhoben werden.

Symbolzuordnung

Zunächst wird beschrieben, welche Ziele im Rahmen der Benutzerstudie verfolgt werden. Durch die gewonnenen Daten soll eine Aussage darüber getroffen werden, welches Verb am besten die durch einen PPE beschriebene Aktion repräsentiert. Des Weiteren soll eine Aussage darüber getroffen werden, in wie weit die Zuordnung intuitiv zu verwenden ist, das heißt in wie weit auch Personen ohne Expertenwissen in der Lage sind, dem Robotersystem auszuführende Aktionen anzuweisen.

Zur Durchführung der Benutzerstudie werden die Probanden in vier Gruppen eingeteilt. Insgesamt werden die Daten von zwölf Probanden ausgewertet, wobei jede Gruppe drei Personen umfasst. Die Einteilung der Gruppen lautet wie folgt:

- *Gruppe A.*
Diese Gruppe umfasst die Personen mit *Allgemeinbildung*. Die Personen dieser Gruppe verfügen insbesondere kein Expertenwissen in den Bereichen Naturwissenschaften, Programmierung oder Robotik. Diese Gruppe stellt somit den Standardbenutzer dar.
- *Gruppe B.*
Diese Gruppe umfasst Personen mit Kenntnissen im Bereich *Naturwissenschaften*, beziehungsweise der Physik. Die Personen dieser Gruppe verfügen insbesondere kein Expertenwissen im Bereich der Programmierung oder der Robotik.
- *Gruppe C.*
Diese Gruppe umfasst Personen mit Kenntnissen im Bereich der *Informatik*, insbesondere in der Softwareentwicklung, beziehungsweise Programmierung. Diese Gruppe verfügt insbesondere kein Expertenwissen im Bereich der Robotik.
- *Gruppe D.*
Diese Gruppe umfasst Personen mit Kenntnissen im Bereich der *Robotik*, insbesondere in der Programmierung von Robotersystemen.

Die Durchführung der Benutzerstudie wird in zwei Bereiche aufgeteilt. Im ersten Teil (Teil A) werden den Probanden zunächst die Ausführungen von diversen PPE vorgeführt. Einige Beispiele sind in Abbildung 3.6 dargestellt. Im Anschluss müssen die Probanden beschreiben, in welcher Form sie die gezeigten Aufgaben verbalisieren, beziehungsweise einem Menschen in Form einer Anweisung beschreiben würden. Hierzu werden keinerlei Einschränkungen an die zu verwendenden Symbole vorgenommen. Durch diesen Teil der Studie wird untersucht, welches Verb zu einem PPE zugeordnet werden soll.

Im zweiten Teil (Teil B) der Benutzerstudie werden den Probanden eine Reihe an Instruktionen vorgegeben, welche zur Ausführung von diversen Aufgaben zur Verfügung stehen. Auch hierbei werden zunächst die auszuführenden Aufgaben demonstriert und im Anschluss die Instruktionen der Probanden notiert. Der Unterschied zum ersten Teil besteht im Wesentlichen in der Einschränkung der zur Verfügung stehenden Aktionen, beziehungsweise Verben. Zur Instruktion der Aufgaben im Rahmen des zweiten Teils der Benutzerstudie stehen die Anweisungen zur Verfügung, welche in Abbildung 3.7 dargestellt sind.

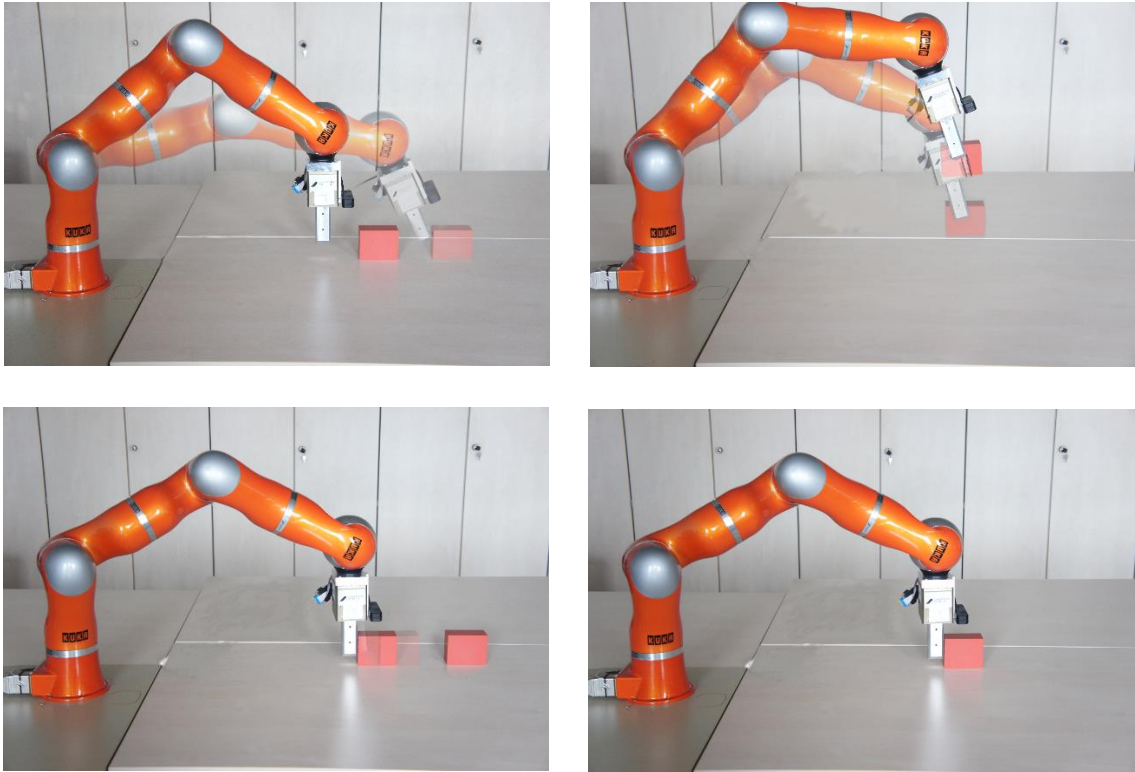


Abbildung 3.6: Beispiele der Ausführungen von PPE, welche den Probanden im Rahmen der Benutzerstudie vorgeführt werden. Links oben (Aufgabe 1) zeigt die Ausführung des Effekts *Transformiere eine Kraft in eine Beschleunigung, bzw. Länge*. Rechts oben (Aufgabe 2) zeigt die Ausführung des Effekts *Ändere die potentielle Energie*. Links unten (Aufgabe 3) zeigt die Ausführung des Effekts *Transformiere einen Impuls in eine Geschwindigkeit, bzw. Länge*. Rechts unten (Aufgabe 4) zeigt die Ausführung des Effekts *Absorbiere eine Kraft*.

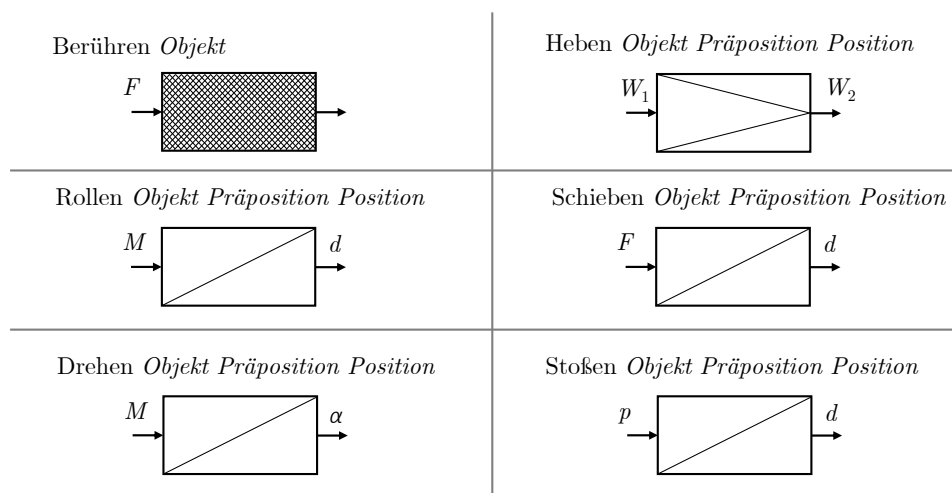


Abbildung 3.7: Verfügbarer Befehlssatz für den zweiten Teil der Benutzerstudie.

	Gruppe A		Gruppe B		Gruppe C		Gruppe D		Total
	Verb	Ratio r _A	Verb	Ratio r _B	Verb	Ratio r _C	Verb	Ratio r _D	Ratio r _T
Aufgabe 1									
Teil A	Schieben	0.67	Schieben	1.00	Schieben	1.00	Schieben	0.67	0.83
	Drücken	0.33					Positionieren	0.33	0.17
Teil B	Schieben	1.00	Schieben	1.00	Schieben	1.00	Schieben	1.00	1.00
Aufgabe 2									
Teil A	Heben	1.00	Heben	1.00	Heben	1.00	Heben	1.00	1.00
Teil B	Heben	1.00	Heben	1.00	Heben	1.00	Heben	1.00	1.00
Aufgabe 3									
Teil A	Stoßen	0.67	Stoßen	1.00	Stoßen	1.00	Stoßen	0.67	0.83
	Drücken	0.33					Schießen	0.33	0.17
Teil B	Stoßen	1.00	Stoßen	1.00	Stoßen	1.00	Stoßen	1.00	1.00
Aufgabe 4									
Teil A	Berühren	0.67	Berühren	0.67	Berühren	1.00	Berühren	0.67	0.75
	Drücken	0.33	Drücken	0.33			Drücken	0.33	0.25
Teil B	Berühren	1.00	Berühren	1.00	Berühren	1.00	Berühren	1.00	1.00

Tabelle 3.8: Ergebnisse der Benutzerstudie.

Die Ergebnisse der Benutzerstudie sind in Tabelle 3.8 dargestellt. In der tabellarischen Übersicht sind die verwendeten Verben der von den Probanden instruierten Anweisungen dargestellt. Die Übersicht ist gruppiert nach der durchzuführenden Aufgabe (Aufgaben siehe Abbildung 3.6) sowie den einzelnen Probandengruppen. Die Ratio r_i mit $i \in \{A, B, C, D, T\}$ beschreibt hierbei das Verhältnis:

$$r_i = \frac{\text{Anzahl der Nennungen eines Verbs } v_n}{\text{Anzahl der Probanden einer Gruppe } |i|}$$

Im Idealfall liegt bei einer entsprechenden Aufgabe $r_T = 1.00$ vor, das heißt alle Probanden verwenden ausschließlich eine eindeutige verbale Beschreibung zur Ausführung eines PPE. Diese kann somit dem entsprechenden PPE zugeordnet werden. Betrachtet man zunächst Teil A der Benutzerstudie, so lässt sich für jede Aufgabe ein von den Probanden bevorzugtes Verb erkennen, was in Form eines dominanten r_T repräsentiert wird. Obwohl keine Einschränkungen an die Instruktionen festgelegt werden, lässt sich bei Aufgabe 2 bereits ein $r_T = 1.00$ beobachten. Für die anderen Aufgaben lässt sich eine dominante, jedoch keine eindeutige verbale Beschreibung zur Ausführung eines PPE feststellen.

Es lässt sich zudem erkennen, dass ein Verb teilweise zur Beschreibung verschiedener PPE verwendet wird. Als Beispiel dient an dieser Stelle das Verb *drücken*, welches bei mehreren Aufgaben genannt wird. Dieses wird auch im allgemeinen Sprachgebrauch nicht eindeutig verwendet. Betrachtet man beispielsweise das Drücken einer Tür, so lässt sich typischerweise eine Bewegung der Tür feststellen, das heißt es wird eine Kraft oder ein Moment in eine Bewegung transformiert. Betrachtet man hingegen das Drücken eines Objekts mit großer Masse, wie beispielsweise einem Auto oder einem Haus, so lässt sich typischerweise keine Bewegung feststellen, das heißt es wird eine Kraft oder ein Moment durch das Objekt absorbiert.

Analysiert man die Ergebnisse von Teil B der Benutzerstudie, so lässt sich feststellen, dass für alle durchgeführten Aufgaben eine eindeutige verbale Beschreibung ermittelt werden kann. Insbesondere sind die Probanden ohne Expertenkenntnisse in der Lage, dem Robotersystem die benötigten Anweisungen zu instruieren. Daraus wird gefolgert, dass die vorgeschlagene Aktionsbeschreibung anhand der PPE als Basis für eine intuitive Benutzerschnittstelle geeignet ist. Die vorgenommenen Einschränkungen an die Instruktionen haben dazu geführt, dass die nicht eindeutige Zuordnung des Verbs *drücken* aufgelöst und ein eindeutiges Verb zur Ausführung eines PPE gefunden werden kann.

Betrachtet man die Valenzen der instruierten Verben, so genügen diese der bereits im Vorfeld geforderten Eigenschaft, $\alpha + \beta + 1$ obligatorische Ergänzungen aufnehmen zu können. Zudem entsprechen die Ergänzungsklassen den erforderlichen Typen, das heißt eine Ergänzung spezifiziert beispielsweise das im Rahmen der Aktion zu manipulierende Objekt, auf welches die Eingangsgröße des PPE entweder aufgebracht oder als Eigenschaft des spezifizierten Objekts berechnet werden kann. Neben dieser ist eine weitere Ergänzung vorhanden, welche die Ausgangsgröße des PPE in symbolischer Form spezifiziert (falls benötigt). Die resultierenden Zusammenhänge für die vier exemplarischen Aufgaben aus der Benutzerstudie sind in Tabelle 3.9 dargestellt. Zur Bestimmung der tatsächlichen Parameter wird zunächst eine geeignete Beschreibung der symbolischen Parameter benötigt, welche im folgenden Kapitel 4 beschrieben wird. Das Auflösen der symbolischen Information sowie die Bestimmung der subsymbolischen Information wird anschließend in Kapitel 5.2, beziehungsweise Kapitel 5.3 beschrieben.

Abschließend lässt sich die Menge der *Verbalisierten Physikalischen Effekte* \mathcal{VPE} definieren. Diese erweitert die Definition der Prinzipiellen Physikalischen Effekte \mathcal{PPE} um eine verbale Komponente, welche auf Basis von empirischen Daten ermittelt und unter Berücksichtigung der Valenz einem entsprechenden PPE zugeordnet wird.

Sei \mathcal{V} die Menge der verfügbaren Verben.

Dann lässt sich die Menge der *Verbalisierten Physikalischen Effekte* \mathcal{VPE} definieren als

$$\mathcal{VPE} := \{ (V, E) \mid V \in \mathcal{V}, E \in \mathcal{PPE} \} \quad (3.8)$$

beziehungsweise

$$\mathcal{VPE} := \{ (V, T, I, O, MPN) \mid V \in \mathcal{V}, T \in \mathcal{PO}, I \in \mathcal{PQ}^\alpha, O \in \mathcal{PQ}^\beta \} \quad (3.9)$$

Aufgabe 1				
Beispielanweisung	Schiebe	den roten Block	30cm nach rechts	
Satzbauplan	NomE	AkkE	AdvE	
Struktur	jemand/etwas	schiebt	jemand/etwas	irgendwohin
Zuordnung		indirekt	direkt	
PPE	(transform,	force	,	length)
Aufgabe 2				
Beispielanweisung	Hebe	den roten Block	30cm nach oben	
Satzbauplan	NomE	AkkE	AdvE	
Struktur	jemand/etwas	hebt	jemand/etwas	irgendwohin
Zuordnung		indirekt	indirekt	
PPE	(change,	energy	,	energy)
Aufgabe 3				
Beispielanweisung	Stoße	den roten Block	nach rechts	
Satzbauplan	NomE	AkkE	AdvE	
Struktur	jemand/etwas	stößt	jemand/etwas	irgendwohin
Zuordnung		indirekt	direkt	
PPE	(transform,	momentum	,	length)
Aufgabe 4				
Beispielanweisung	Berühre	den roten Block		
Satzbauplan	NomE	AkkE		
Struktur	jemand/etwas	berührt	jemand/etwas	
Zuordnung		indirekt		
PPE	(absorb,	force	,	∅)

Tabelle 3.9: Qualitative Zusammenhänge zwischen symbolischen Anweisungen, Satzbauplänen, Strukturen und Parameter des zu einem Verb zugeordneten PPE. Bei den verwendeten Beispielanweisungen handelt es sich um konkrete Anweisungen, welche im Rahmen der Benutzerstudie durch Probanden instruiert wurden.

3.4 Zusammenfassung

Im Rahmen dieses Kapitels wurde die wissenschaftliche Fragestellung betrachtet, welche die systematische Beschreibung von ausführbaren Roboteraktionen adressiert. Diese stellt eine Teilmenge der im Kapitel 1.2 festgelegten wissenschaftlichen Fragestellungen dar und lautet wie folgt:

- F1** In welcher Form kann die Semantik einer ausführbaren Aktion systematisch beschrieben und in einem Robotersystem hinterlegt werden?

Um die Semantik von ausführbaren Aktionen zu beschreiben, musste zunächst eine geeignete Zwischenrepräsentation gefunden werden. Diese wird benötigt, da im Allgemeinen keine direkte Zuordnung zwischen symbolischer und subsymbolischer Information möglich ist. Hierzu wurde die Verwendung von physikalischen Effekten, beziehungsweise physikalischen Gesetzmäßigkeiten, Größen und Nebenbedingungen vorgeschlagen. Diese Art der Zwischenrepräsentation scheint geeignet, da die physikalischen Gesetze sowohl für die symbolische Repräsentation des Menschen, als auch für die subsymbolische Repräsentation des Robotersystems gültig und im Allgemeinen nicht außer Kraft gesetzt werden können.

Aufbauend auf den physikalischen Bausteinen wurde ein Konzept zur systematischen Beschreibung von Roboteraktionen vorgeschlagen. Hierzu wurde zunächst das Konzept der Prinziplösungen aus dem Bereich der Systementwicklung für die Verwendung von physikalischen Größen adaptiert. Daraus resultieren eine Reihe von physikalischen Operationen auf physikalische Größen. Basierend auf diesen wurde das Konzept der *Prinzipiellen Physikalischen Effekte* eingeführt, welches die Zusammenhänge zwischen der gewählten Zwischenrepräsentation und der subsymbolischen Repräsentation des Robotersystems beschreibt. Im Rahmen dieser Arbeit entspricht die subsymbolische Repräsentation der Definition von Aktionsprimitiven, beziehungsweise Aktionsprimitivnetzen. Daher wurden zunächst die Zusammenhänge zwischen Prinzipiellen Physikalischen Effekten, physikalischen Gesetzmäßigkeiten und physikalischen Nebenbedingungen beschrieben. Zudem wurden die Zusammenhänge zwischen Prinzipiellen Physikalischen Effekten und dem zugeordneten Aktionsprimitivnetz dargestellt. Dies umfasste die Beschreibung der Struktur und die Parametrisierung des zu einem Prinzipiellen Physikalischen Effekt zugeordneten Aktionsprimitivnetzes.

Zur Beschreibung der Zusammenhänge zwischen der gewählten Zwischenrepräsentation und der symbolischen Repräsentation sowie des Einflusses der symbolischen Parameter auf die auszuführende Aktion wurde das Konzept erweitert zu *Verbalisierten Physikalischen Effekten*. Hierzu wurde beschrieben, in welcher Form die Zuordnung eines Verbs zu einer Aktionsbeschreibung basierend auf einem PPE umgesetzt werden kann, sowie in welcher Form die zusätzlichen Angaben einer symbolischen Anweisung die Ausführung, beziehungsweise die Parametrisierung einer auszuführenden Aktion beeinflussen.

Abschließend sind die Zusammenhänge, welche im Rahmen dieses Kapitels erarbeitet und beschrieben wurden, in schematischer Form in Abbildung 3.8 dargestellt.

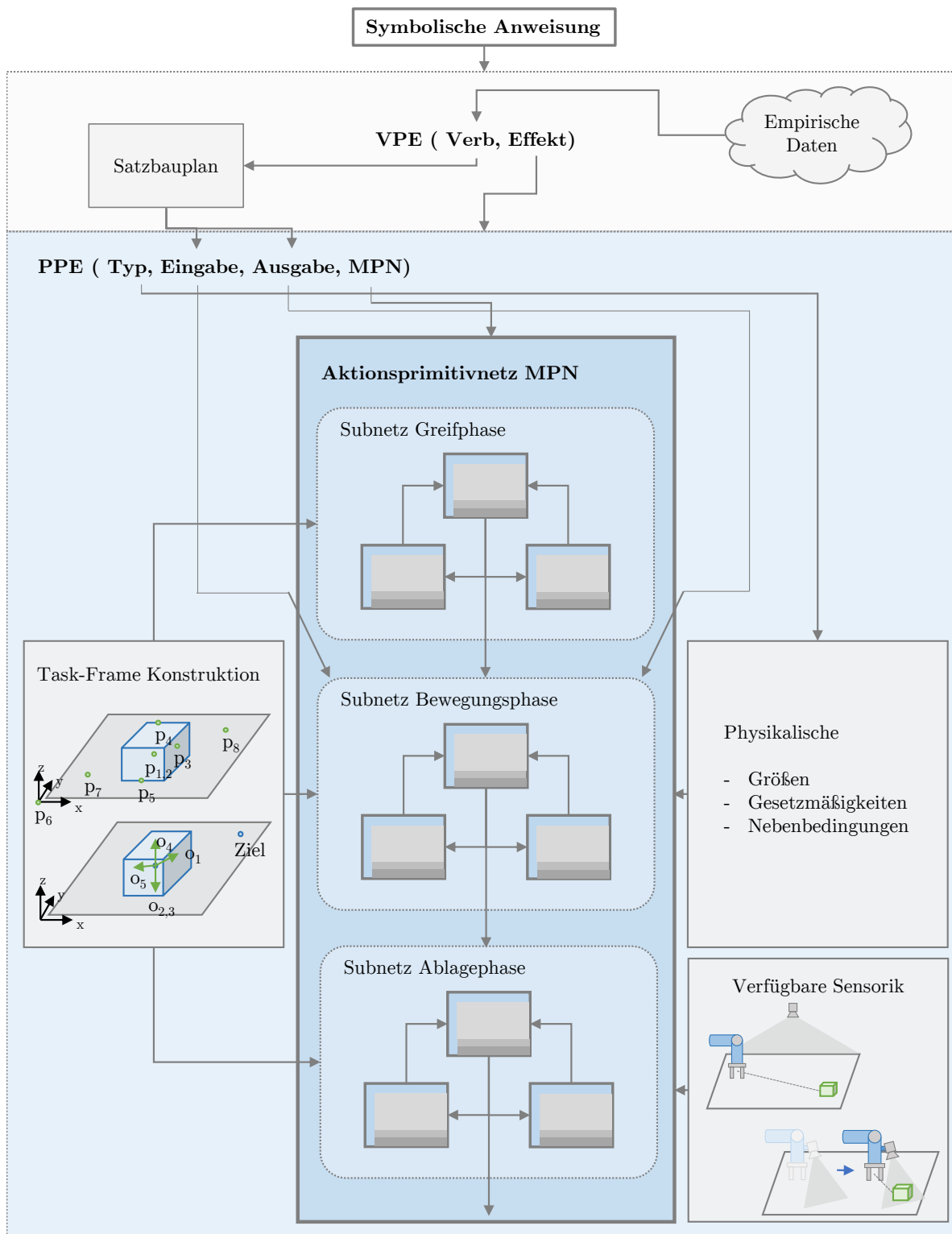


Abbildung 3.8: Schematische Zusammenhänge der im Rahmen dieses Kapitels erarbeiteten und beschriebenen Konzepte.

4 Hinterlegen von Symbolen zur Roboterkommandierung

Eine auszuführende Anweisung besteht neben der Spezifikation der Aktion, beziehungsweise des Verbs aus einer Reihe weiterer symbolischer Parameter, welche die zu einem Verb benötigten Ergänzungen in einer geeigneten Ausdrucksform beschreiben (siehe Kapitel 3.3). Im Rahmen dieses Kapitels wird daher untersucht, in welcher Form diese zusätzlichen Symbole beschrieben und im Robotersystem hinterlegt werden können. Dies entspricht im Wesentlichen der Fragestellung **F2**, welche wie folgt lautet:

- F2** In welcher Form kann die Semantik der zur Parametrisierung einer ausführbaren Aktion verwendeten Symbole systematisch beschrieben und in einem Robotersystem hinterlegt werden?

Wie in Kapitel 3.2 beschrieben, verwendet die vorgeschlagene Aktionsbeschreibung physikalische Größen als Ein- und Ausgabeparameter, welche auf Basis der zur symbolischen Parametrisierung einer Anweisung verwendeten Symbole extrahiert werden sollen. Daraus folgt, dass diese symbolischen Parameter auf Basis von sensorisch erfassbaren, physikalischen Größen beschrieben werden müssen. Zum einen ist dies nötig, um gemeinsam mit der vorgeschlagenen Aktionsbeschreibung im Rahmen eines Gesamtsystems verwendet zu werden. Zum anderen ist dies nötig, um die automatisierte Informationsextraktion durch das Robotersystem zu gewährleisten. Daher wird im Rahmen dieses Kapitels untersucht, welchen physikalischen Informationsgehalt die verschiedenen Typen von Symbolen besitzen und in welcher Form diese Information im Robotersystem hinterlegt werden kann.

Abschließend wird die Struktur dieses Kapitels beschrieben. Zunächst werden die zur symbolischen Parametrisierung von verbalisierten Anweisungen verwendeten Symbole kategorisiert. Zudem wird untersucht, in welcher Form die verschiedenen Symbole im allgemeinen Sprachgebrauch verwendet werden sowie in welcher Form diese die symbolische Anweisung beeinflussen, beziehungsweise welchen Informationsgehalt die einzelnen Symbole besitzen (Kapitel 4.1). Aufbauend auf diesen Erkenntnissen wird beschrieben, in welcher Form die Symbolinformation im Robotersystem hinterlegt werden kann (Kapitel 4.2). Abschließend werden die gewonnenen Erkenntnisse zusammengefasst und diskutiert (Kapitel 4.3).

Die im folgenden dargestellte Analyse sowie das Hinterlegen der Symbolinformation ist im Rahmen von [Spangenberg16b] veröffentlicht.

4.1 Kategorien von Symbolen

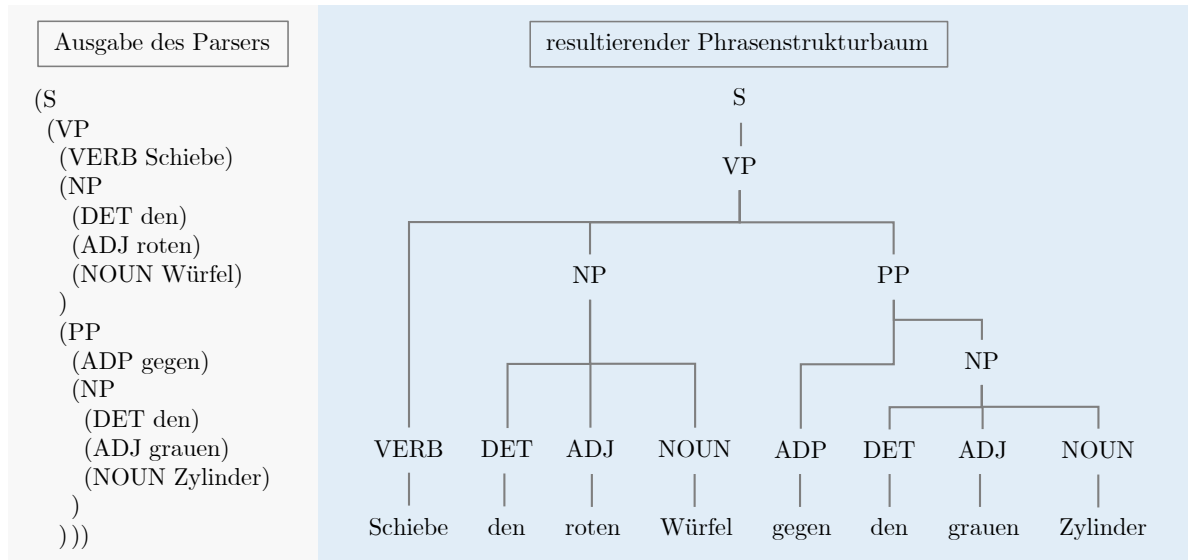
Wie bereits im Rahmen von Kapitel 3.3 beschrieben und in Tabelle 3.6 dargestellt, beschreiben die zusätzlich zu einem Verb instruierten Symbole die obligatorischen oder fakultativen Ergänzungen. Die hierbei verwendeten Symbole werden bei semantisch vollständigen und syntaktisch korrekten Anweisungen nicht willkürlich aneinander gereiht, sondern in Form von geeigneten Satzbausteinen formuliert. Diese Satzbausteine werden im Allgemeinen als *Phrase* oder *syntaktische Einheit* bezeichnet. Bevor die einzelnen Symbole beschrieben und kategorisiert werden, werden zunächst die verschiedenen Satzbausteine erläutert. Zur Kategorisierung wird die Beschreibung nach [Taylor03] verwendet, welche unter anderem zur Annotation der Inhalte in der *PennTreebank* [Marcus93] eingesetzt wird. Die wesentlichen Phrasen nach [Taylor03] lauten wie folgt:

- *Satz* (S)
- *Verbalphrase* (VP)
- *Adjektivphrase* (ADJP)
- *Adverbialphrase* (ADVP)
- *Nominalphrase* (NP)
- *Präpositionalphrase* (PP)
- *sonstige* (X)

Diese unterschiedlichen Phrasen sind aus verschiedenen Symbolen zusammengesetzt, welche typischerweise auf Basis der *Wortart* (part-of-speech, PoS) kategorisiert werden können. An dieser Stelle werden die Wortarten basierend auf einer universellen Beschreibung nach [Petrov11] kategorisiert. Hierbei werden insgesamt zwölf Kategorien von Symbolen beschrieben. Zudem wird gezeigt, dass auf Basis dieser Kategorien ein Großteil der Texte aus der englischsprachigen *PennTreebank* [Marcus93] (ca. 97 %) oder deutschsprachigen Corpora wie *Tiger* [Brants02] oder *Negra* [Skut97] (ca. 98 %) annotiert werden kann. Die im Rahmen von [Petrov11] vorgenommene Unterteilung lautet wie folgt:

- *Nomen* (NOUN)
- *Verben* (VERB)
- *Adjektive* (ADJ)
- *Adverbien* (ADV)
- *Pronomen* (PRON)
- *Artikel* (DET)
- *Prä- und Postpositionen* (ADP)
- *Numerale* (NUM)
- *Konjunktionen* (CONJ)
- *Partikel* (PART)
- *Satzzeichen* (.)
- *sonstige* (X)

Die Annotation der Struktur einer Anweisung sowie die Klassifizierung der Wortart eines Symbols wird typischerweise durch einen *Parser* wie beispielsweise [Klein03] durchgeführt. Die Annotation der Beispielanweisung *Schiebe den roten Würfel gegen den blauen Zylinder* ist in Abbildung 4.1 dargestellt.



Abbildungung 4.1: Exemplarische Annotation der Struktur einer Anweisung. Die Ausgabe wird durch einen Parser nach [Klein03] beziehungsweise unter Berücksichtigung der universellen Beschreibung nach [Petrov11] erzeugt.

Im Folgenden werden die einzelnen Eigenschaften der Kategorien von Symbolen beschrieben. Dies umfasst zum einen eine Beschreibung der prinzipiellen Funktion im Sprachgebrauch nach [Kessel12]. Zum anderen wird betrachtet, in welchen Phrasen die entsprechenden Symbolarten typischerweise verwendet werden sowie in wie weit diese Abhängigkeiten zu weiteren Symbolen besitzen. Dies stellt die allgemeinen Zusammenhänge dar, welche bereits in angrenzenden Forschungsbereichen wie der Linguistik, beziehungsweise der Computerlinguistik untersucht wurden und im Rahmen dieser Arbeit an geeigneter Stelle verwendet werden (siehe Kapitel 5.2).

Zusätzlich zu den allgemeinen Zusammenhängen wird untersucht, welche relevante subsymbolische Information anhand der verschiedenen Symboltypen extrahiert werden kann. Diese muss zusätzlich zu den allgemeinen Zusammenhängen im Robotersystem hinterlegt werden und bei der Parametrisierung einer auszuführenden Roboteraktion genutzt werden, was im Rahmen von Kapitel 5 beschrieben wird.

Ein Symbol aus der Gruppe der *Nomen* (NOUN) wird typischerweise im Rahmen einer Nominalphrase verwendet und beschreibt im Allgemeinen einen Gattungsnamen (Appellativa) oder Eigennamen (nomina propria). Die Gattungsnamen werden weiter unterteilt in konkrete Nomen (Konkreta) und abstrakte Nomen (Abstrakta). Die konkreten Nomen beschreiben hierbei entweder bestimmte Objekte (*Messer*, *Gabel*) oder Gruppen von Objekten (*Besteck*) der realen Welt. Die abstrakten Nomen beschreiben hingegen Eigenschaften (*Freundlichkeit*, *Tapferkeit*), Vorgänge (*Auszahlung*, *Altern*) oder Zustände (*Krieg*, *Frieden*).

Des Weiteren lassen sich die Symbole dieser Gruppe anhand des Numerus unterteilen, wobei typischerweise zwischen singular und plural unterschieden wird. Zum einen unterscheidet man an dieser Stelle zwischen Symbolen, welche für beide Ausprägungen ein identisches Symbol verwenden (beispielsweise *das Messer* - *die Messer*). Zum anderen existieren Symbole, welche für jede Ausprägung ein jeweils unterschiedliches Symbol verwenden (beispielsweise *die Gabel* - *die Gabeln*). Im erstgenannten Fall lässt sich eine Abhängigkeit zu den Symbolen aus der Gruppe der Artikel feststellen, da diese beschreiben, welche Form des Symbols im Rahmen der Anweisung verwendet wird. Betrachtet man die für das Robotersystem relevante Symbolinformation, so beschreibt ein Nomen entweder ein konkretes, reales Objekt oder eine Gruppe von realen Objekten, welche jeweils eine Menge von individuellen physikalischen Eigenschaften wie eine Position, Masse oder Farbe besitzen. Der Numerus des Symbols beinhaltet zudem Information darüber, wie häufig eine entsprechende Aktion ausgeführt werden muss. Dies erfordert gegebenenfalls eine zusätzliche Komponente im Robotersystem, welche die Ausführung koordiniert.

Ein Symbol aus der Gruppe der *Verben* (VERB) wird typischerweise im Rahmen einer Verbalphrase verwendet und beschreibt im Allgemeinen eine Tätigkeit (*singen, lachen, tippen*), einen Vorgang (*einschlafen, wachsen, verblühen*) oder einen Zustand (*stehen, liegen, wohnen*). Wie bereits in Kapitel 3.3 dargestellt, beschreibt das Verb einer Anweisung im Allgemeinen die auszuführende Aktion. Zudem sind auf Basis dieses Symbols die zur semantisch vollständigen Beschreibung benötigten obligatorischen Ergänzungen sowie die hierzu erforderlichen Phrasen, beziehungsweise Ausdrucksformen spezifiziert. Durch die im Rahmen dieser Arbeit vorgeschlagene Aktionsbeschreibung wird durch dieses Symbol somit festgelegt, welcher Verbalisierte Physikalische Effekt ausgeführt werden soll und welche physikalischen Parameter zur Ausführung durch das Robotersystem benötigt werden.

Ein Symbol aus der Gruppe der *Adjektive* (ADJ) wird im Rahmen einer Nominal-, Adjektiv- oder Adverbialphrase verwendet und beschreibt eine Eigenschaft eines Objekts oder einer Aktion. Je nach Verwendung lässt sich eine Abhängigkeit zu einem Nomen (das *schnelle* Auto) oder einem Verb (das Auto fährt *schnell*) feststellen. Im Allgemeinen können diese Symbole daher die physikalischen Eigenschaften von Objekten oder Aktionen genauer spezifizieren, beispielsweise geometrische (der *große* Würfel), kinematische (das *schnelle* Auto) oder dynamische Eigenschaften (der *schwere* Würfel). Zudem können diese beispielsweise visuelle Eigenschaften spezifizieren (der *rote* Würfel). Des Weiteren besitzen einige Symbole dieser Gruppe die Eigenschaft der Komparation, welche typischerweise unterteilt wird in Positiv (*groß*), Komparativ (*größer*) und Superlativ (am *größten*). Im Unterschied zu Nomen, welche als Numerus ausschließlich ein einzelnes Objekt oder eine Vielzahl von Objekten beschreiben, sind Adjektive in der Lage, Intervallgrenzen (*minimal, maximal*) oder eine exakte Anzahl (*zweifach, dreifach*) zu spezifizieren. Des Weiteren lässt sich feststellen, dass die Adjektive eine Objekteigenschaft im Allgemeinen nicht durch einen exakten subsymbolischen Wert beschreiben, sondern in Form einer unscharfen Beschreibung spezifizieren. Betrachtet man beispielsweise das Adjektiv *grün*, so beschreibt dies eine visuelle Eigenschaft eines Objekts, welcher eine Menge von Farbwerten eines entsprechenden Farbraums zugeordnet werden kann.

Daraus folgt, dass ein verwendetes Adjektiv typischerweise den erlaubten Wertebereich einer physikalischen Eigenschaft spezifiziert, beziehungsweise einschränkt. Im Allgemeinen bedingt dies somit einen Vergleich zwischen einem aktuellen, sensorisch erfassten Istwert und einem im System hinterlegten Sollwert.

Ein Symbol aus der Gruppe der *Adverbien* (ADV) wird typischerweise im Rahmen einer Adverbialphrase verwendet. Diese Gruppe von Symbolen wird im Allgemeinen unterteilt in Temporaladverbien (*jetzt, danach*), Lokaladverbien (*rechts, mittig*), Modaladverbien (*gern, größtenteils*), Kausaladverbien (*deshalb, weshalb*) und Negationsadverbien (*nicht, keinesfalls*). Typischerweise werden diese verwendet, um eine Eigenschaft eines Objekts oder einer Aktion genauer zu spezifizieren. Ein Temporaladverb spezifiziert beispielsweise Information über die Priorität oder die Reihenfolge von auszuführenden Aufgaben (*Zuerst* greife den Würfel! *Danach* greife den Zylinder!). Dies kann somit eine zusätzliche Komponente zur Aufgabenkoordination erfordern. Ein Lokaladverb spezifiziert typischerweise eine geometrische Eigenschaft wie beispielsweise eine Ortsangabe oder eine Richtung (Berühre den Würfel *oben rechts*!). Da es sich hierbei im Allgemeinen um relative Angaben handelt, muss hierzu das zur relativen Angabe verwendete Koordinatensystem bekannt sein. Ist dieses bekannt, können beispielsweise Symbole wie *rechts* oder *oben* aufgelöst werden. Des Weiteren existieren Symbole, welche zusätzliche Information zur Auflösung benötigen (*hier, da, dorthin*). Diese erfordern beispielsweise eine zusätzliche Komponente in Form einer Gestenerkennung.

Ein Symbol aus der Gruppe der *Pronomen* (PRON) wird typischerweise im Rahmen einer Nominalphrase verwendet. Die Gruppe der Symbole wird im Allgemeinen unterteilt in Personalpronomen (*ich, du*), Demonstrativpronomen (*dieser, jener*), Possessivpronomen (*mein, dein*), Interrogativpronomen (*was, welcher*), Reflexivpronomen (*welcher, der ...*) und Indefinitpronomen (*man, jeder, keiner*). Diese Symbole werden typischerweise dazu verwendet, ein Symbol vom Typ NOUN zu ersetzen. Daher ist eine Abhängigkeit zwischen diesen beiden Symboltypen festzustellen. Die Zuordnung des zu einem Pronomen zugeordnete Nomen stellt eine der klassischen Aufgaben der Sprachverarbeitung dar, welche als *entity resolution* oder *co-reference resolution* bezeichnet wird [Sammut11]. Da diese Symbole auf ein Nomen verweisen, entspricht der Informationsgehalt dem des referenzierten Nomens.

Ein Symbol aus der Gruppe der *Artikel* (DET) wird typischerweise im Rahmen einer Nominalphrase verwendet. Wie bereits beschrieben, lässt sich hierbei eine Abhängigkeit zu dem im Rahmen der Nominalphrase spezifizierten Nomen feststellen, für welches der Artikel den Genus, den Numerus sowie die Bestimmtheit festlegt. Die Gruppe wird im Allgemeinen unterteilt in definite (*der, die, das*) und indefinite (*ein, eine*) Artikel.

Je nach gewähltem Symbol kann der Numerus des zugeordneten Nomens festgelegt werden. Dies kann unter Umständen zu einer widersprüchlichen Aussage führen, welche beispielsweise in Form einer Nachfrage aufgelöst werden muss. In einem weiteren Fall kann dies beispielsweise dazu führen, dass eine einzelne Anweisung eine mehrfache Ausführung durch das Robotersystem zur Folge hat. Ein Beispiel hierzu ist in Abbildung 4.2 dargestellt.

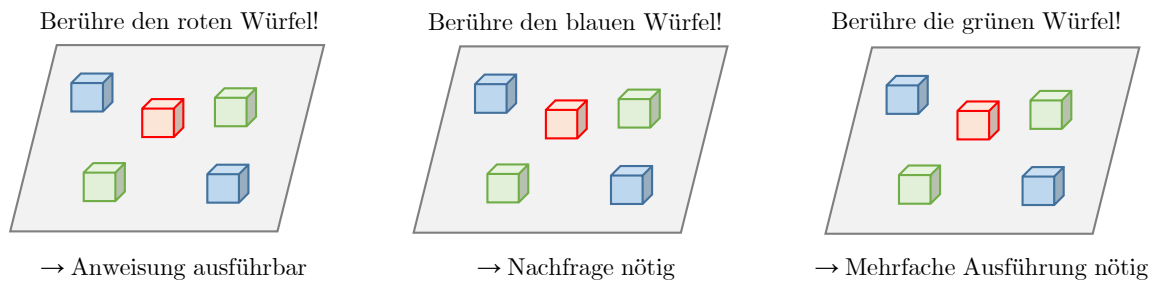


Abbildung 4.2: Einfluss des Artikels auf die Ausführung einer Beispielanweisung.

Die Gruppe der *Prä- und Postpositionen* (ADP) wird typischerweise im Rahmen von Präpositionalphrasen verwendet und beschreiben im Allgemeinen Relationen zwischen Objekten. Diese werden unterteilt in lokal (auf, neben), temporal (vor, nach), modal (mit, gemäß) und kausal (aufgrund, wegen). Die Präpositionen stehen hierbei typischerweise vor dem Objekt, beziehungsweise Satzglied, auf welches sie sich beziehen (*auf dem Tisch!*). Die Postpositionen stehen typischerweise nach dem Objekt, beziehungsweise Satzglied (der Oberfläche *entlang*). Betrachtet man beispielsweise die Gruppe der lokalen Präpositionen, so können diese als eine Relation auf physikalischen Größen beschrieben werden (siehe Gleichung 3.5).

Die Gruppe der *Konjunktionen* (CONJ) wird typischerweise unterteilt in kopulativ (und, wie), disjunkt (oder), restriktiv (aber, nur) und kausal (denn). Im Allgemeinen beschreiben diese Symbole eine Relation zwischen Sätzen, Nebensätzen, Satzgliedern oder einzelnen Worten. Wird eine Konjunktion verwendet, um einen Zusammenhang zwischen Sätzen herzustellen, erfordert dies gegebenenfalls eine Komponente, welche die Ausführung koordiniert (Greife den Würfel *und* schiebe ihn gegen den Zylinder!). Wird eine Konjunktion zur Beschreibung eines Zusammenhangs zwischen Worten beschrieben (Greife den roten oder blauen Würfel!) so entspricht dies ähnlich wie bei den lokalen Präpositionen einer Relation auf physikalischen Größen.

Betrachtet man abschließend die Gruppe der *Numerale* (NUM), so beschreiben diese eine subsymbolische Information exakt und in direkter Form (eins, fünf). Diese kann prinzipiell jeder physikalischen Größe zugeordnet werden. Dies bedingt, dass sich ein Symbol dieses Typs auf ein anderes Symbol beziehen muss, welches die relevante physikalische Größe spezifiziert (in fünf Sekunden).

Basierend auf der vorangegangenen Beschreibung der verschiedenen Kategorien von Symbolen, lassen sich eine Reihe an Schlussfolgerungen extrahieren, welche für die Ausführung und Parametrisierung von symbolischen Anweisungen relevant sind. Diese Zusammenhänge können im Vorfeld im Rahmen einer Wissensbasis im Robotersystem hinterlegt, beziehungsweise zur Laufzeit durch das Robotersystem genutzt werden. Zunächst lässt sich eine Reihe an allgemeinen Zusammenhängen festhalten. Diese resultieren in erster Linie aus der Struktur der verbal instruierten Anweisungen und lauten wie folgt:

- Die verschiedenen Symbole lassen sich auf Basis der Wortart kategorisieren.
- Die Symbole werden im Rahmen von Satzbauteilen verwendet.
- Die Satzbauteile beschreiben die notwendigen Ergänzungen zu einem Verb.
- Es existieren Abhängigkeiten zwischen Symbolgruppen, welche auf Basis der Satzbauteile aufgelöst werden können.
- Die Annotation der Struktur und der Symbole wird durch einen Parser realisiert.

Neben den allgemeinen Zusammenhängen können zudem eine Reihe an Zusammenhängen extrahiert werden, welche den Einfluss verschiedener Symbole auf die Parametrisierung und Ausführung einer verbal instruierten Anweisung beschreiben. An dieser Stelle ist in erster Linie relevant, in welcher Form die Symbole die zur Ausführung benötigten physikalischen Eigenschaften beeinflussen. Die Zusammenhänge zwischen den symbolischen Parametern und physikalischen Eigenschaften lauten wie folgt:

- Ein Symbol kann ein reales Objekt beschreiben, welches eine Reihe an physikalischen Eigenschaften besitzt.
- Ein Symbol kann einen ausführbaren Prozess beschreiben, welcher einem Verbalisierten Physikalischen Effekt zugeordnet wird und eine Reihe an physikalischen Eigenschaften besitzt.
- Ein Symbol kann eine Relation zwischen Objekten oder Aktionen, beziehungsweise deren physikalischen Eigenschaften beschreiben.
- Ein Symbol kann physikalische Eigenschaften von Objekten oder ausführbaren Aktionen beschreiben.
- Ein Symbol kann eine physikalische Eigenschaft auf verschiedene Arten spezifizieren.

Diese Zusammenhänge werden im Rahmen einer Wissensbasis im Robotersystem hinterlegt und zur Ausführung, beziehungsweise Parametrisierung von symbolisch instruierten Anweisungen verwendet. Die im Rahmen dieser Arbeit vorgeschlagene Komponente zum Hinterlegen der Symbolinformation wird im nächsten Kapitel beschrieben.

4.2 Physikalisches Wörterbuch

Eine der zusätzlichen Komponenten, welche ein Robotersystem zur Parametrisierung und Ausführung von symbolisch instruierten Anweisungen benötigt, stellt eine Wissensbasis dar. Diese beschreibt unter anderem die Zusammenhänge zwischen den symbolischen Parametern der Anweisung sowie den dadurch spezifizierten subsymbolischen Parametern, welche zur Ausführung der Anweisung benötigt werden. Im Rahmen dieser Arbeit wird hierzu ein *physikalisches Wörterbuch* vorgeschlagen, welches zwei grundsätzliche Informationen zur Verfügung stellt:

- Die Beschreibung der Symbolsemantik im Kontext von spezifizierten physikalischen Größen, beziehungsweise physikalischen Gesetzmäßigkeiten.
- Die Beschreibung von Sensoren und Komponenten, welche die zur Ausführung benötigten physikalischen Größen aus der Umgebung extrahieren können.

Die erste Information basiert auf der im Rahmen von Kapitel 4.1 dargestellten Analyse der verschiedenen Kategorien von Symbolen. Dies umfasst zum einen die allgemeine Symbolinformation, welche auf Basis der Struktur der verbal instruierten Anweisung extrahiert werden kann. Zum anderen umfasst dies die physikalische Beschreibung der Symbolsemantik, welche zur Nutzung der im Rahmen dieser Arbeit vorgeschlagenen Aktionsbeschreibung benötigt wird.

Die zweite Information ist relevant zur automatischen Parametrisierung der ausführbaren Aktion. Wie in Kapitel 1.3 dargestellt, stehen dem Robotersystem eine Reihe an Komponenten zur Verfügung, welche in der Lage sind, bestimmte physikalische Eigenschaften aus der Umgebung zu extrahieren. Diese Komponenten nutzen hierfür typischerweise geeignete Sensorik.

Da eine physikalische Eigenschaft jedoch entweder direkt durch Extraktion der entsprechenden Größe oder indirekt über eine physikalische Gesetzmäßigkeit extrahiert werden kann (siehe Kapitel 3.3), lassen sich eine Reihe weiterer Zusammenhänge festhalten, welche im Rahmen des physikalischen Wörterbuchs berücksichtigt werden müssen. Diese lauten wie folgt:

- Eine physikalische Größe kann entweder direkt oder indirekt bestimmt werden.
- Die indirekte Bestimmung einer gesuchten physikalischen Größe erfolgt durch die direkte Bestimmung von physikalischen Größen, welche durch eine physikalische Gesetzmäßigkeit in Verbindung gebracht werden können.
- Die direkte Bestimmung einer physikalischen Größe erfolgt durch eine Extraktionskomponente.
- Die Extraktionskomponente verwendet zur Extraktion geeignete Sensorik.

Daraus resultieren die im Rahmen des physikalischen Wörterbuchs hinterlegten Zusammenhänge, welche in Abbildung 4.3 dargestellt sind. Zur Umsetzung des physikalischen Wörterbuchs stehen verschiedene Möglichkeiten zur Verfügung, beispielsweise in Form einer relationalen Datenbank, einer Ontologie oder einer logikbasierten Repräsentation. Im Rahmen des prototypischen Gesamtsystems ist das physikalische Wörterbuch in Form einer relationalen Datenbank umgesetzt (siehe Kapitel 6.1).

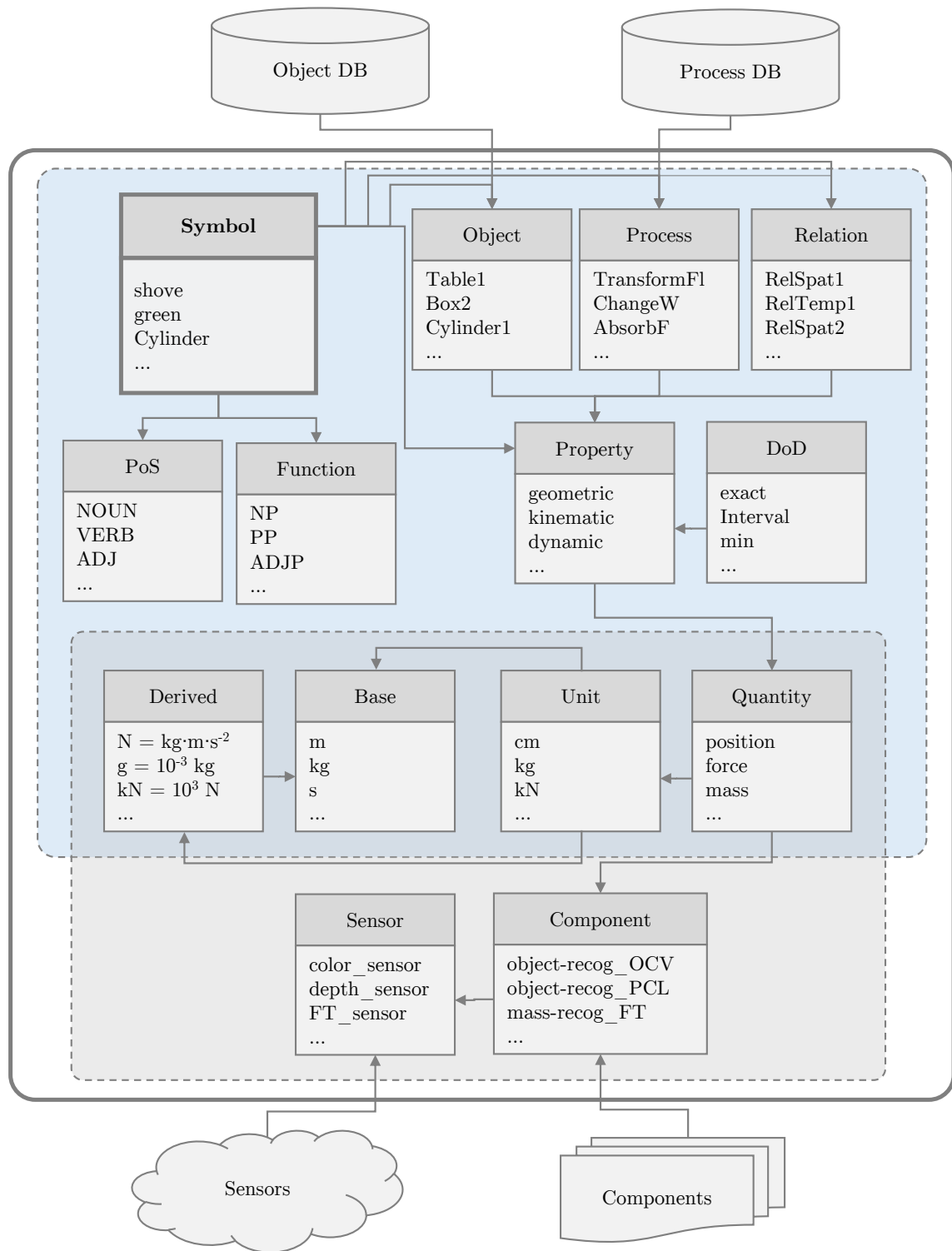


Abbildung 4.3: Schematische Darstellung der im Rahmen des physikalischen Wörterbuchs hinterlegten Information (weiße Umrandung). Dies umfasst zum einen Information zur Beschreibung der Symbolsemantik in allgemeiner Form sowie im Kontext der physikalischen Eigenschaften (blaue Umrandung). Zum anderen umfasst dies Information zur automatischen Parametrisierung und Extraktion von physikalischen Eigenschaften (graue Umrandung). Zudem sind eine Reihe externer Komponenten vorhanden. Dies sind beispielsweise die Sensorik des Robotersystems oder die Komponenten zur Extraktion der physikalischen Eigenschaften.

Abschließend werden einige Beispieldatensätze aus dem physikalischen Wörterbuch beschrieben (siehe Abbildung 4.4 und Abbildung 4.5), welche der im Rahmen der prototypischen Umsetzung (siehe Kapitel 6.1) verwendeten Wissensbasis entnommen sind.

----- General Symbol Information -----	----- General Symbol Information -----
Symbol: cube Synonym: - Type: Noun Tag: NOUN Function: NP	Symbol: red Synonym: - Type: Adjective Tag: ADJ Function: ADJP,NP
----- Symbol Semantics Information -----	----- Symbol Semantics Information -----
Describes: Object DB: cube.obj	Describes: Object->Property DB: -
Property/Quantity: geometric/position DoD: exact Unit: XYZABC Value: - // more object properties...	Property/Quantity: visual/color DoD: interval Unit: HSV Value: (0 70 45) - (10 100 50) (355 50 50) - (359 100 70)
Property/Quantity: dynamic/mass DoD: exact Unit(derived): kg Value: -	
----- Parameter Extraction Information -----	----- Parameter Extraction Information -----
Property/Quantity: geometric/position Component: object-recog_OCV Sensor: color_sensor // more extraction components...	Property/Quantity: visual/color Component: color-recog_OCV Sensor: color_sensor
Property/Quantity: dynamic/mass Component: mass-recog_FT Sensor: FT_sensor	

Abbildung 4.4: Exemplarische Einträge des physikalischen Wörterbuchs.

Betrachtet man die im System hinterlegte Information für das Symbol *cube*, so beschreibt die allgemeine Symbolinformation die Wortart (Nomen) sowie die typische Verwendung des Symbols (Nominalphrase). Dieses Symbol beschreibt ein reales Objekt, welches eine Reihe an physikalischen Eigenschaften besitzt. Im Beispiel sind zwei Eigenschaften dargestellt, zum einen eine geometrische Eigenschaft *position* und eine dynamische Eigenschaft *mass*. Da die Werte dieser Eigenschaften nicht im Vorfeld festgelegt sind, müssen diese zur Ausführung durch das Robotersystem aus der Umgebung extrahiert werden. Hierzu wird im System hinterlegt, welche Komponenten zur Extraktion einer physikalischen Eigenschaft genutzt werden können. Im Beispiel wird die Extraktion der Position des Objekts durch eine Komponente *object-recog_OCV* unter Nutzung eines *color_sensor* und die Extraktion der Masse des Objekts durch eine Komponente *mass-recog_FT* unter Nutzung eines *FT_sensor* umgesetzt werden.

Betrachtet man die im System hinterlegte Information für das Symbol *red*, so beschreibt dieses eine Eigenschaft des Objekts, welche im Rahmen der Phrase referenziert wird. Wie bereits im Rahmen von Kapitel 4.1 beschrieben, wird durch ein Symbol vom Typ Adjektiv im Allgemeinen eine physikalische Eigenschaft nicht exakt, sondern gemäß eines Wertebereichs spezifiziert. Im Rahmen der prototypischen Umsetzung (siehe Kapitel 6.1) wird dieser durch Angabe eines Intervalls spezifiziert. Dieses muss bei Ausführung einer Anweisung überprüft werden. Im Beispiel wird hierzu eine Komponente *color-recog_OCV* unter Nutzung eines *color_sensor* verwendet.

----- General Symbol Information -----		----- General Symbol Information -----	
Symbol:	the	Symbol:	a
Synonym:	-	Synonym:	an
Type:	Determiner	Type:	Determiner
Tag:	DET	Tag:	DET
Function:	NP, PP	Function:	NP, PP
----- Symbol Semantics Information -----		----- Symbol Semantics Information -----	
Describes:	Object->Property	Describes:	Object->Property
DB:	-	DB:	-
Property/Quantity:	frequency/amount	Property/Quantity:	frequency/amount
DoD:	exact	DoD:	min
Unit:	piece	Unit:	piece
Value:	1	Value:	1
----- Parameter Extraction Information -----		----- Parameter Extraction Information -----	
Property/Quantity:	frequency/amount	Property/Quantity:	frequency/amount
Component:	object-recog_OCV	Component:	object-recog_OCV
Sensor:	color_sensor	Sensor:	color_sensor

Abbildung 4.5: Exemplarische Einträge des physikalischen Wörterbuchs.

Weitere Symbole zur Beschreibung einer Objekteigenschaft sind beispielsweise die verschiedenen Arten von Artikeln (siehe Abbildung 4.5). Während der bestimmte Artikel *the* die Anzahl des im Rahmen der Phrase spezifizierten Objekts auf genau ein Objekt festlegt, beschreibt der unbestimmte Artikel *a* beziehungsweise *an* ausschließlich die Minimalanzahl des betreffenden Objekts. Zur Bestimmung der Anzahl der Objekte steht auch hier eine Extraktionskomponente zur Verfügung. Im Beispiel wird hierzu ebenfalls die Komponente *color-recog_OCV* unter Nutzung eines *color_sensor* verwendet. Diese wird beispielsweise auch genutzt um die geometrische Eigenschaft *position* eines Objekts zu bestimmen (siehe Abbildung 4.4). Daher kann eine Komponente gegebenenfalls zur Extraktion verschiedener physikalischer Eigenschaften genutzt werden. Der Einfluss der im System verfügbaren Extraktionskomponenten auf die Ausführung einer symbolischen Anweisung wird im Rahmen von Kapitel 6.2 untersucht.

4.3 Zusammenfassung

Im Rahmen dieses Kapitels wurde die wissenschaftliche Fragestellung **F2** betrachtet, welche die Beschreibung der symbolischen Parameter adressiert. Diese lautet wie folgt:

- F2** In welcher Form kann die Semantik der zur Parametrisierung einer ausführbaren Aktion verwendeten Symbole systematisch beschrieben und in einem Robotersystem hinterlegt werden?

Zur Beschreibung der symbolischen Parameter einer Anweisung wurden zunächst die verschiedenen Klassen der Symbole kategorisiert. Hierzu wurde untersucht, in welcher Form die verschiedenen Arten von Symbolen im Rahmen von symbolischen Anweisungen verwendet werden sowie welche allgemeinen Zusammenhänge daraus extrahiert und zur Ausführung durch das Robotersystem genutzt werden können. Des Weiteren wurde untersucht, welchen Informationsgehalt die verschiedenen Kategorien der Symbole im Kontext auf die im Rahmen dieser Arbeit vorgeschlagene Aktionsbeschreibung auf Basis der physikalischen Eigenschaften besitzen.

Zum Hinterlegen der Symbolinformation wurde die Nutzung eines physikalischen Wörterbuchs vorgeschlagen. Diese Komponente beschreibt zum einen die Symbolsemantik im Kontext der durch ein Symbol spezifizierten physikalischen Größen. Zum anderen stellt diese Komponente die Information zur Verfügung, welche Komponenten zur Extraktion von physikalischen Größen durch das Robotersystem genutzt werden. Das physikalische Wörterbuch und die entsprechenden Extraktionskomponenten stellen somit eine Reihe zusätzlicher Komponenten dar, welche einem Robotersystem zur Ausführung von symbolisch instruierten Anweisung zur Verfügung stehen müssen.

5 Parametrisierung von Roboteraktionen unter Nutzung hinterlegter Symbolinformation

Im Rahmen dieses Kapitels wird beschrieben, in welcher Form die in Kapitel 3 vorgeschlagene Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten zusammen mit der in Kapitel 4 vorgeschlagenen physikalischen Beschreibung von Symbolen zur Parametrisierung von ausführbaren Roboteraktionen auf Basis von Aktionsprimitivnetzen verwendet werden kann. Dies entspricht im Wesentlichen der wissenschaftlichen Fragestellung **F3**:

F3 In welcher Form können die vorgeschlagenen Konzepte zur Beschreibung der Symbolsemantik gemeinsam verwendet werden um eine subsymbolische Parametrisierung der auszuführenden Roboteraktionen zu ermöglichen?

Wie bereits in Kapitel 1.3 beschrieben, wird die Extraktion der zur Ausführung benötigten, subsymbolischen Parameter durch eine Reihe dem Robotersystem zur Verfügung stehenden Extraktionskomponenten ausgeführt. Daher wird im Rahmen dieses Kapitels beschrieben, in welcher Form die Symbolbeschreibung und die im System verfügbaren Extraktionskomponenten im Rahmen eines integrierten Gesamtsystems verwendet werden können. Des Weiteren wird beschrieben, in welcher Form eine symbolische Benutzerschnittstelle auf Basis der vorgeschlagenen Konzepte umgesetzt werden kann. Dies stellt die konzeptionelle Grundlagen für die Umsetzung und Evaluation eines prototypischen Gesamtsystems zur Verfügung, was im Rahmen von Kapitel 6 dargestellt wird.

Abschließend wird die Struktur dieses Kapitels beschrieben. Zunächst wird erläutert, in welcher Form die beschriebenen Konzepte zur Umsetzung einer intuitiven, symbolischen Benutzerschnittstelle verwendet und im Robotersystem hinterlegt werden können. Hierzu wird die Nutzung der Konzepte im Rahmen einer domänenspezifischen Sprache vorgeschlagen (Kapitel 5.1). Diese wird anschließend verwendet, um die weiteren Schritte hin zur Ausführung der symbolisch instruierten Anweisung zu beschreiben. Hierbei besteht ein Schritt darin, die symbolische Information der instruierten Anweisung unter Nutzung der Verbalisierten Physikalischen Effekte sowie des physikalischen Wörterbuchs aufzulösen (Kapitel 5.2). Basierend darauf wird beschrieben, wie die zur Ausführung benötigte, subsymbolische Information unter Nutzung von geeigneten Komponenten und Sensoren aus der Umgebung extrahiert und das auszuführende Aktionsprimitivnetz parametrisiert werden kann (Kapitel 5.3). Abschließend werden die gewonnenen Erkenntnisse zusammengefasst und diskutiert (Kapitel 5.4).

Die im Folgenden beschriebenen Prozesse und verwendeten Komponenten sind im Rahmen von [Spangenberg16a, Spangenberg16b, Werner16] veröffentlicht.

5.1 Domänenspezifische Sprache

Eine Möglichkeit zur gemeinsamen Nutzung der vorgeschlagenen Konzepte stellt die Verwendung im Rahmen einer domänenspezifischen Sprache dar. Weitere Möglichkeiten sind beispielsweise die Nutzung im Rahmen eines symbolischen Planungssystems oder im Rahmen einer natürlichsprachlichen Benutzerschnittstelle (siehe Kapitel 7.2).

Bei einer domänenspezifischen Sprache handelt es sich im Allgemeinen um eine Programmier- oder Spezifikationssprache, welche für eine abgegrenzte Problemdomäne eine angemessene Notation und Abstraktion zur Verfügung stellt [VanDeursen00]. Die Abstraktion und Notation soll zudem für alle beteiligten Benutzergruppen möglichst intuitiv vorgenommen werden [Voelter13].

Für den Bereich der Robotik lässt sich eine steigende Nutzung von domänenspezifischen Sprachen beobachten, was im Rahmen von [Nordmann14] beschrieben wird. Neben der Beschreibung vorhandener Arbeiten werden diese auf Basis des Verwendungszwecks der domänenspezifischen Sprache kategorisiert. Die Einsatzgebiete von domänenspezifischen Sprachen nach [Nordmann14] ist in Abbildung 5.1 dargestellt.

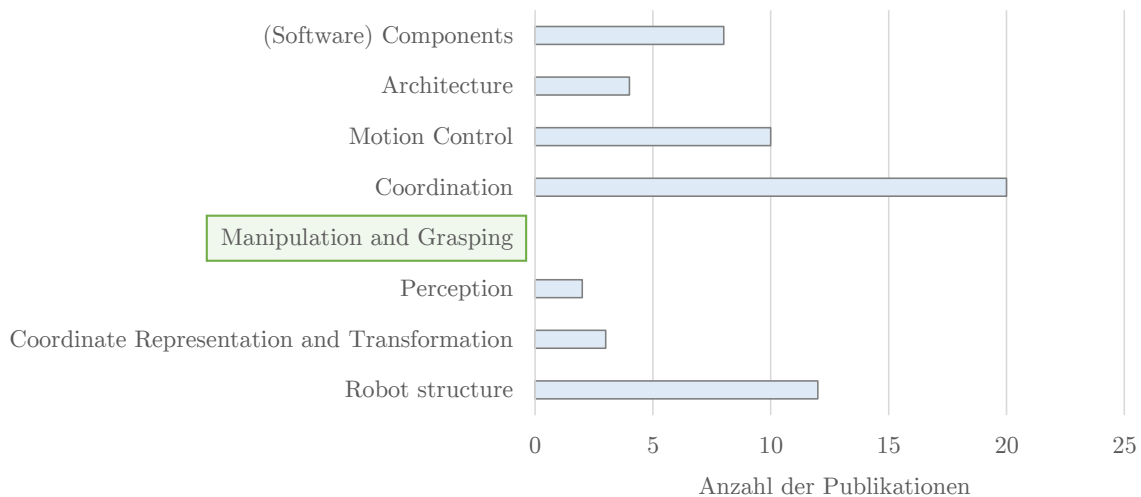


Abbildung 5.1: Kategorisierung der Einsatzgebiete und Einordnung der betrachteten Arbeiten, welche domänenspezifische Sprachen im Bereich der Robotik verwenden. Die Kategorien und Einordnung sind aus [Nordmann14] entnommen.

Wie in Abbildung 5.1 ersichtlich, wird keine der in [Nordmann14] betrachteten Arbeiten dem Bereich der Objektmanipulation zugeordnet. Ein Grund hierfür kann wie in Kapitel 2.4 beschrieben eine fehlende systematische Beschreibung der Zusammenhänge zwischen einer symbolischen Anweisung, einer subsymbolischen Aktionsbeschreibung und der Parametrisierung der auszuführenden Roboteraktion darstellen.

Die auszuführenden Aktionen stellen typischerweise Schlüsselworte der domänenspezifischen Sprache dar, welche mit einer entsprechenden Semantik verknüpft und im Vorfeld in einer parametrisierbaren Form im Robotersystem hinterlegt werden. Dies kann unter Nutzung der in Kapitel 3 und 4 vorgeschlagenen Konzepte umgesetzt werden.

Erste Arbeiten betrachten die Nutzung von domänenspezifischen Sprachen für Aufgaben aus dem Bereich der Objektmanipulation [Buchmann13]. Diese beschränken sich auf Pick&Place Aufgaben, welche auf elementare Roboteraktionen abgebildet werden. Durch die im Rahmen dieser Arbeit vorgeschlagene Aktionsbeschreibung kann die Menge der ausführbaren Aufgaben auf sensorbasierte Aktionen erweitert werden.

Zunächst wird die zur Spezifikation von validen Anweisungen gemäß der domänenspezifischen Sprache verwendete Notation sowie die vorgenommene Abstraktion beschrieben. Die Schlüsselworte der domänenspezifischen Sprache entsprechen der Menge der verbalen Ausdrücke, welche im Rahmen der Verbalisierten Physikalischen Effekte einem entsprechenden Effekt zugeordnet werden. Dies entspricht somit Ausdrücken wie beispielsweise *bump*, *rotate* oder *shove*.

Zur Spezifikation der Parameter stehen verschiedene Ebenen der Abstraktion zur Verfügung. So können beispielsweise die physikalischen Eigenschaften des entsprechenden Effekts direkt als Parameterwerte genutzt werden. Bei der Aktion *bump*, welche dem prinzipiellen Effekt

$$PPE = (transform, (A, Nm, torque), (B, m, length))$$

zugeordnet wird (siehe Kapitel 3.2), können die Parameter *A* und *B* direkt in Form von subsymbolischen Werten spezifiziert werden. Dies entspricht jedoch nicht der im Rahmen der Aufgabenstellung geforderten symbolischen, intuitiv zu verwendenden Benutzerschnittstelle. Daher werden die Parameter in einer symbolischen Darstellung instruiert, welche unter Nutzung zusätzlicher Komponenten in die subsymbolische Darstellung überführt wird. Im Rahmen der domänenspezifischen Sprache werden hierzu die obligatorischen Ergänzungen des verbalen Ausdrucks verwendet, welche in Form von geeigneten Phrasen und unter Nutzung der im Rahmen des physikalischen Wörterbuchs verfügbaren Symbole instruiert werden.

Betrachtet man exemplarisch den verbalen Ausdruck *bump*, so besitzt dieser drei obligatorische Ergänzungen. Eine Ergänzung beschreibt den ausführenden Agenten der Anweisung, welcher implizit dem Befehlsempfänger - und daher dem Robotersystem - zugeordnet wird und im Allgemeinen bei Befehlsanweisungen nicht instruiert wird. Die weiteren Ergänzungen spezifizieren das zu manipulierende Objekt, beispielsweise im Form einer Nominalphrase sowie die gewünschte Positionsänderung, beispielsweise in Form einer Präpositionalphrase (siehe Kapitel 3.3).

Im Allgemeinen entspricht die Struktur einer Anweisung somit der Form:

Aktion *Parameterphrase₁* ... *Parameterphrase_n*

Die zu einer Aktion relevanten Daten werden im Rahmen eines XML-Schemas spezifiziert. Dieses beschreibt zum einen die Information, welche zur Definition von gültigen Anweisungen gemäß der domänenspezifischen Sprache benötigt wird. Zum anderen beschreibt dieses die zu einem Schlüsselwort zugeordnete Semantik, welche aus dem zugeordneten Verbalisierten Physikalischen Effekt, beziehungsweise dem zur Ausführung verwendeten Aktionsprimitivnetz besteht. Im Folgenden werden die relevanten Abschnitte des XML-Schemas dargestellt und beschrieben.

Das Wurzelement stellt die Beschreibung der Struktur eines Verbalisierten Physikalischen Effekts **VPE** dar, welcher sich aus einer verbalen Komponente **VERB**, einer optionalen Menge an Synonymen **SYN** sowie dem zum Verbalisierten Physikalischen Effekt zugeordneten Prinzipiellen Physikalischen Effekt **TYPE** und einem Aktionsprimitivnetz **MPN** zusammensetzt (siehe Code 5.1).

```
<!-- Verbalized Physical Effect (VPE) -->
<xs:element name="VPE">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="VERB" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <xs:element name="SYN" type="xs:string" minOccurs="0" maxOccurs="3"/>
      <xs:choice>
        <xs:element ref="ABSORB_TYPE"/>
        <xs:element ref="CHANGE_TYPE"/>
        <xs:element ref="MERGE_TYPE"/>
        <xs:element ref="SPLIT_TYPE"/>
        <xs:element ref="TRANSFORM_TYPE"/>
      </xs:choice>
      <xs:element ref="MPN"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Code 5.1: Ausschnitt der XML-Schema Definition zur Spezifikation der Verbalisierten Physikalischen Effekte.

Der nächste Abschnitt beschreibt die Struktur der verschiedenen Effekttypen, deren Parameter sowie die zur symbolischen Instruktion verwendeten Ausdrucksformen. Dies beinhaltet zum einen die Definition und Zuordnung der Eingabeparameter IN und Ausgabeparameter OUT eines Verbalisierten Physikalischen Effekts (siehe Code 5.2).

```
<!-- Specification of effect parameter types-->
<xs:complexType name="TYPE_BASE">
  <xs:sequence>
    <xs:element ref="IN" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element ref="OUT" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- Absorb: 1 in, 0 out -->
<xs:complexType name="TYPE_1x0">
  <xs:complexContent>
    <xs:restriction base="TYPE_BASE">
      <xs:sequence>
        <xs:element ref="IN" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="OUT" minOccurs="0" maxOccurs="0"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<!-- Change, Transform: 1 in, 1 out -->
<xs:complexType name="TYPE_1x1">
  <xs:complexContent>
    <xs:restriction base="TYPE_BASE">
      <xs:sequence>
        <xs:element ref="IN" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="OUT" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<!-- ... Specification of other effect parameter types -->

<!-- Assignment of parameter types to effect types -->
<xs:element name="ABSORB_TYPE" type="TYPE_1x0"/>
<xs:element name="CHANGE_TYPE" type="TYPE_1x1"/>
<xs:element name="MERGE_TYPE" type="TYPE_Ax1"/>
<xs:element name="SPLIT_TYPE" type="TYPE_1xN"/>
<xs:element name="TRANSFORM_TYPE" type="TYPE_1x1"/>
```

Code 5.2: Ausschnitt der XML-Schema Definition zur Spezifikation der Ein- und Ausgabeparameter der Verbalisierten Physikalischen Effekte.

Zum anderen beinhaltet dies die Spezifikation der einzelnen physikalischen Parameter des Effekts (**propertyType** beziehungsweise **quantityType**) inklusive der im Rahmen der domänenspezifischen Sprache zu verwendenden Ausdrucksform einer Ergänzung **phrase** (siehe Code 5.3).

```
<!-- Input and Output Parameters with physical quantities -->
<xs:simpleType name="IN_OUT_VALUE_TYPE">
  <xs:restriction base="xs:string">
    <!-- List of physical quantities (excerpt) -->
    <xs:pattern value="acceleration|angle|angular|area|..."/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="IN_OUT_TYPE">
  <xs:simpleContent>
    <xs:extension base="IN_OUT_VALUE_TYPE">
      <xs:attribute name="phrase">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <!-- List of syntactical functions based on [Taylor03] -->
            <xs:enumeration value="VP"/>
            <xs:enumeration value="NP"/>
            <xs:enumeration value="PP"/>
            <xs:enumeration value="ADJP"/>
            <xs:enumeration value="ADVP"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>

      <xs:attribute name="propertyType">
        <!-- ... Specification of physical properties -->
      </xs:attribute>
      <xs:attribute name="quantityType">
        <!-- ... Specification of physical quantities -->
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="IN" type="IN_OUT_TYPE"/>
<xs:element name="OUT" type="IN_OUT_TYPE"/>
```

Code 5.3: Ausschnitt der XML-Schema Definition zur Spezifikation der physikalischen Eigenschaften sowie den im Rahmen der symbolischen Eingabe verwendeten Phrasen für die entsprechenden Ein- und Ausgabeparameter der Verbalisierten Physikalischen Effekte.

Die Instanzen, welche gemäß des beschriebenen Schemas spezifiziert werden, entsprechen den ausführbaren Verbalisierten Physikalischen Effekten und somit den Schlüsselworten, welche im Rahmen der domänenspezifischen Sprache verwendet werden können. Im Folgenden wird für jeden Effekttyp ein Beispiel für einen zugeordneten Verbalisierten Physikalischen Effekt sowie dessen Nutzung im Rahmen der domänenspezifischen Sprache dargestellt. Die Zuordnung der verbalen Beschreibung zu einem physikalischen Effekt ist auf Basis von empirischen Daten ermittelt worden, was im Rahmen von Kapitel 3.3 beschrieben ist.

Als Beispiel für einen Effekt vom Typ *absorb* dient der prinzipielle Effekt

$$PPE = (absorb, (A, N, force), \emptyset)$$

welchem die verbale Beschreibung *touch* zugeordnet wird. Wie in Kapitel 3.3 beschrieben, handelt es sich hierbei um ein zweiwertiges Verb, welches somit zwei obligatorische Ergänzungen benötigt. Eine Ergänzung entspricht dem Akteur der Instruktion, welcher bei Befehlen typischerweise nicht explizit formuliert, sondern implizit dem Befehlsempfänger zugeordnet wird. Im Rahmen der domänenspezifischen Sprache entspricht der Akteur dem Robotersystem. Die zweite obligatorische Ergänzung spezifiziert das zu manipulierende Objekt, welches im Allgemeinen durch eine Nominalphrase beschrieben wird. Im Kontext der Verbalisierten Physikalischen Effekte entspricht dies somit dem Objekt, auf welches die Eingangsgrößen des Effekts angewandt, beziehungsweise basierend auf dessen physikalischen Eigenschaften bestimmt werden. Das Verfahren zur Bestimmung und Extraktion der physikalischen Eigenschaften unter Nutzung des physikalischen Wörterbuchs sowie der im System verfügbaren Extraktionskomponenten wird im Rahmen von Kapitel 5.3 beschrieben.

Auf Basis der verbalen Komponente, der allgemeinen Ausdrucksform der Ergänzung sowie des zugeordneten Prinzipiellen Physikalischen Effekts resultiert die in Code 5.4 dargestellte Beschreibung für die Aktion *touch*.

```
<verbbot:VPE
  xmlns:verbbot="http://www.ai3.uni-bayreuth.de/projects/verbbot"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.ai3.uni-bayreuth.de/projects/verbbot./vpe.xsd">

  <verbbot:VERB> touch </verbbot:VERB>

  <verbbot:ABSORB_TYPE>
    <verbbot:IN phrase="NP" propertyType="dynamic" quantityType="force">
      forceParam
    </verbbot:IN>
  </verbbot:ABSORB_TYPE>

  <!-- ... Specification of the manipulation primitive net -->

</verbbot:VPE>
```

Code 5.4: Ausschnitt der XML-Aktionsdefinition zur Spezifikation der Aktion *touch*.

Dieser Abschnitt spezifiziert somit eine Anweisung der Form

touch *Nominalphrase*

welche beispielsweise die Instruktion von symbolischen Anweisungen wie

touch *the red box*

touch *one of the blue cylinders*

erlaubt. Die Analyse der Anweisung wird unter Nutzung eines Parsers wie beispielsweise [Klein03] durchgeführt, welches einen Vorgang im Rahmen der Auflösung der symbolischen Information darstellt. Das Verfahren zum Auflösen der symbolischen Information wird im nächsten Kapitel 5.2 beschrieben.

Als Beispiel für einen Effekt vom Typ *change* dient der prinzipielle Effekt

$$PPE = (change, (W_1, J, energy), (W_2, J, energy))$$

welchem je nach gewählter physikalischer Nebenbedingung (siehe Kapitel 3.2) entweder die verbale Beschreibung *lift* oder *sink* zugeordnet wird. Hierbei handelt es sich um einen dreiwertigen verbalen Ausdruck, welcher neben der Angabe des Akteurs, der implizit dem Robotersystem zugeordnet wird das zu manipulierende Objekt sowie die gewünschte Änderung beschreibt. Diese werden typischerweise im Rahmen einer Nominal- beziehungsweise einer Präpositionalphrase instruiert (siehe Kapitel 3.3). Daraus resultiert die Effektbeschreibung, welche in Code 5.5 dargestellt ist.

```
<verbbot:VPE
  xmlns:verbbot="http://www.ai3.uni-bayreuth.de/projects/verbbot"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.ai3.uni-bayreuth.de/projects/verbbot_./vpe.xsd">

  <verbbot:VERB> lift </verbbot:VERB>

  <verbbot:CHANGE_TYPE>
    <verbbot:IN phrase="NP" propertyType="dynamic" quantityType="energy">
      energyParam1
    </verbbot:IN>
    <verbbot:OUT phrase="PP" propertyType="dynamic" quantityType="energy">
      energyParam2
    </verbbot:OUT>
  </verbbot:CHANGE_TYPE>

  <!-- ... Specification of the manipulation primitive net -->

</verbbot:VPE>
```

Code 5.5: Ausschnitt der XML-Aktionsdefinition zur Spezifikation der Aktion *lift*.

Dieser Abschnitt spezifiziert somit eine Anweisung der Form

lift *Nominalphrase* *Präpositionalphrase*

welche beispielsweise die Instruktion von folgenden symbolischen Anweisungen erlaubt:

lift *the red box* *for 30 cm*

lift *the red box* *on top of the blue cube*

Zum Abschluss wird ein Beispiel für einen Effekt vom Typ *transform* betrachtet. Hierzu wird der Effekt

$$PPE = (transform, (F, N, force), (l, m, length))$$

verwendet, welchem der verbale Ausdruck *shove* zugeordnet wird. Auch hierbei handelt es sich um einen dreiwertigen Ausdruck, welcher neben dem Akteur der Anweisung das zu manipulierende Objekt sowie die gewünschte Änderung als obligatorische Ergänzungen besitzt (siehe Kapitel 3.3). Diese werden typischerweise im Rahmen einer Nominal- beziehungsweise einer Präpositionalphrase instruiert. Daraus resultiert die Effektbeschreibung, welche in Code 5.6 dargestellt ist.

```
<verbbot:VPE
  xmlns:verbbot="http://www.ai3.uni-bayreuth.de/projects/verbbot"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.ai3.uni-bayreuth.de/projects/verbbot_1./vpe.xsd">

  <verbbot:VERB> shove </verbbot:VERB>

  <verbbot:TRANSFORM_TYPE>
    <verbbot:IN phrase="NP" propertyType="dynamic" quantityType="force">
      forceParam
    </verbbot:IN>
    <verbbot:OUT phrase="PP" propertyType="geometric" quantityType="length">
      lengthParam
    </verbbot:OUT>
  </verbbot:TRANSFORM_TYPE>

  <!-- ... Specification of the manipulation primitive net -->

</verbbot:VPE>
```

Code 5.6: Ausschnitt der XML-Aktionsdefinition zur Spezifikation der Aktion **shove**.

Dieser Abschnitt spezifiziert somit eine Anweisung der Form

shove *Nominalphrase* *Präpositionalphrase*

welche beispielsweise die Instruktion von folgenden symbolischen Anweisungen erlaubt:

shove *the red box* *towards the blue cube*

shove *the red box* *for 5 cm to the right*

Abschließend lässt sich festhalten, dass die vorgeschlagenen Konzepte gemeinsam im Rahmen einer domänenspezifischen Sprache verwendet werden können. Die domänenspezifische Sprache nutzt die Menge der zu den entsprechenden Verbalisierten Physikalischen Effekten zugeordneten verbalen Beschreibungen als Schlüsselworte und deren obligatorische Ergänzungen als Parameter. Diese können dem System unter Nutzung der Symbole instruiert werden, welche im Rahmen des physikalischen Wörterbuchs zur Verfügung stehen. Die Struktur der Anweisungen sowie die Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten werden im Rahmen einer XML-basierten Beschreibung im Robotersystem hinterlegt. Da die Aktionsbeschreibung unabhängig von der Roboterkonfiguration spezifiziert wird, kann diese für verschiedene Robotersysteme mit unterschiedlicher Sensorik eingesetzt werden, was im Rahmen von Kapitel 6.2 dargestellt wird. In wie weit die vorgeschlagene domänenspezifische Sprache als intuitive Benutzerschnittstelle verwendet werden kann, wird im Rahmen von Kapitel 6.3 evaluiert.

5.2 Auflösen von symbolischer Information

Ein Schritt hin zur Ausführung der instruierten Anweisung stellt das Auflösen der symbolischen Information dar. Dies umfasst zunächst die Identifikation der auszuführenden Aktion. Hierzu wird anhand der verbalen Beschreibung - beziehungsweise im Kontext der domänenspezifischen Sprache anhand des Schlüsselwortes - die zur Instruktion zugehörige XML-basierte Aktionsbeschreibung gesucht und geladen. Ab dieser Stelle ist bekannt, welche obligatorischen Ergänzungen die entsprechende Instruktion benötigt, sowie welche subsymbolischen Parameter diese besitzt.

Nach erfolgreicher Identifikation der auszuführenden Aktion wird die instruierte Anweisung unter Nutzung eines Parsers [Klein03] verarbeitet. Im Rahmen des Verarbeitungsschrittes wird der sogenannte Phrasenstrukturbaum konstruiert, welcher die instruierte Anweisung in eine Baumstruktur überführt (siehe Abbildung 5.2). Auf Basis des resultierenden Phrasenstrukturbaums wird die Validierung der Instruktion vorgenommen. Hierzu wird die Information aus der XML-basierten Aktionsbeschreibung mit der Struktur der instruierten Anweisung abgeglichen.

Im nächsten Schritt werden die einzelnen Symbole der Instruktion analysiert und mit der im Rahmen des physikalischen Wörterbuchs hinterlegten Symbolinformation annotiert. Der abschließende Schritt besteht im Auflösen von Abhängigkeiten. Diese sind beispielsweise zwischen den im Rahmen der obligatorischen Ergänzungen spezifizierten Objekte und den Ein- beziehungsweise Ausgangsgrößen des auszuführenden Verbalisierten Physikalischen Effekt als auch zwischen verschiedenen Symbolen innerhalb einer Parameterphrase vorhanden.

Eingabe: symbolische Instruktion I
Ausgabe: annotierter Phrasenstrukturbaum O

```

1  function resolve_symbolic_information(I)
2      action = identify_action_and_parameter(I)
3      if (action != null)
4          O = parse_instruction(I)
5          if (O != null)
6              ret = validate_instruction(action, O)
7              if (ret)
8                  foreach symbol in O
9                      if (in_dictionary(symbol))
10                         O.at(symbol) = append_dictionary_information(symbol)
11                     else
12                         return null          /* ERROR_unknown_symbol */
13                     end
14                 end
15                 if (!resolve_references(action, O))
16                     return null          /* ERROR_resolve_references */
17                 end
18             else
19                 return null          /* ERROR_invalid_structure */
20             end
21         else
22             return null          /* ERROR_could_not_parse_instruction */
23         end
24     else
25         return null          /* ERROR_unknown_action */
26     end
27     return O
28 end function

```

Algorithmus 5.1: Verfahren zum Auflösen von symbolischer Information einer instruierten Anweisung.

Anhand der beschriebenen Operationen lässt sich ein Algorithmus zum Auflösen der symbolischen Information einer instruierten Anweisungen herleiten (siehe Algorithmus 5.1). Die Eingabe des Algorithmus stellt die durch den Benutzer instruierte, symbolische Anweisung dar, welche im Rahmen des Algorithmus in einen annotierten Phrasenstrukturbaum überführt wird, in dem sowohl die Struktur der Anweisung als auch die physikalische Symbolinformation zu den entsprechenden Symbolen der Instruktion repräsentiert sind. Im Folgenden werden die einzelnen Schritte des Algorithmus anhand eines Beispiels im Detail beschrieben.

Betrachtet man exemplarisch die Anweisung

shove the red cube towards a blue cylinder

so besteht der erste Schritt des Algorithmus (siehe Zeile 2, Algorithmus 5.1) in der Identifikation sowie dem Laden der auszuführenden Aktion. Für die Beispielanweisung wird die XML-basierte Aktionsbeschreibung genutzt, welche in Code 5.6 dargestellt ist. Auf Basis der Aufgabenbeschreibung steht somit die Information über die benötigten Parameterphrasen, die Ein- und Ausgabeparameter des zum verbalen Ausdruck zugeordneten physikalischen Effekts sowie dem zur Ausführung der Aktion hinterlegten Aktionsprimitivnetz zur Verfügung.

Im nächsten Schritt des Algorithmus (siehe Zeile 4, Algorithmus 5.1) wird die instruierte Anweisung durch einen Parser verarbeitet. Dies beinhaltet die Annotation von genereller Symbolinformation zu den einzelnen Symbolen als auch die Repräsentation der Anweisungsstruktur in Form eines Phrasenstrukturbaums. Der zur Beispielanweisung konstruierte Phrasenstrukturbaum ist in Abbildung 5.2 dargestellt.

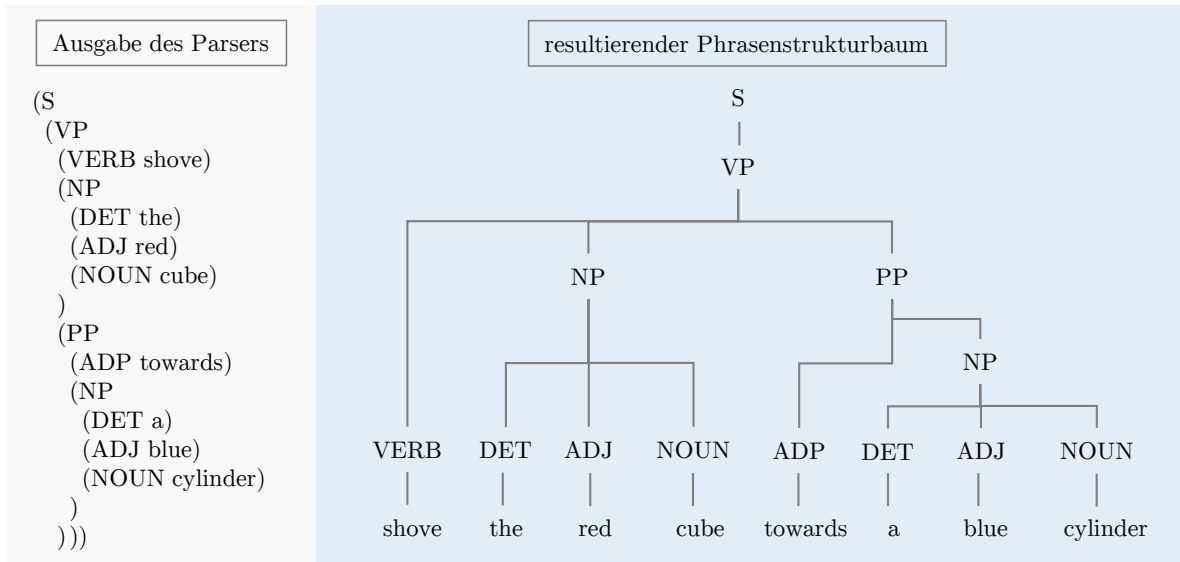


Abbildung 5.2: Resultierender Phrasenstrukturbaum für eine Beispielanweisung.

Dieser wird verwendet, um die Struktur der instruierten Anweisung zu validieren (siehe Zeile 6, Algorithmus 5.1). Hierzu wird geprüft, ob die im Rahmen der XML-basierten Aktionsbeschreibung spezifizierten obligatorischen Ergänzungen in der erwarteten Ausdrucksform vorhanden sind. Für die Beispielanweisung wird eine Nominal- und eine Präpositionalphrase erwartet, um eine semantisch vollständige und korrekte Anweisung gemäß der domänenspezifischen Sprache zu formulieren. An dieser Stelle wird genutzt, dass die Anweisungen der domänenspezifischen Sprache auf Basis der obligatorischen Ergänzungen eines verbalen Ausdrucks festgelegt werden, wodurch im Phrasenstrukturbaum einer gültigen Anweisung eine Verbalphrase vorhanden ist, welche die entsprechenden Ergänzungen als eingebettete Phrasen besitzt. Betrachtet man den Phrasenstrukturbaum der Beispielanweisung (siehe Abbildung 5.2) so besitzt die Anweisung eine Verbalphrase sowie die geforderten Ergänzungen in den gewünschten Ausdrucksformen als eingebettete Phrasen, wodurch die instruierte Anweisung der im Rahmen der XML-basierten Aktionsbeschreibung spezifizierten Struktur genügt.

Im nächsten Schritt (siehe Zeile 8-10, Algorithmus 5.1) wird zu jedem Symbol des Phrasenstrukturbaums die im physikalischen Wörterbuch hinterlegte Symbolinformation (siehe Abbildung 4.4) annotiert. Diese umfasst insbesondere die Beschreibung der Symbolsemantik im Kontext der spezifizierten physikalischen Eigenschaften, die Abhängigkeiten zwischen Symbolen als auch die Information über die zur Extraktion der physikalischen Eigenschaften benötigten Extraktionskomponenten, welche zur Parametrisierung des auszuführenden Aktionsprimitivnetzes benötigt werden (siehe Kapitel 5.3).

In der exemplarischen Anweisung umfasst dies beispielsweise die Annotation des Symbols `cube`, welches diverse physikalische Eigenschaften wie eine Position, Farbe oder Masse besitzt (siehe Abbildung 5.3). Wird für jedes verwendete Symbol ein Eintrag im physikalischen Wörterbuch gefunden, so kann die instruierte Anweisung als valide Instruktion gemäß der domänenspezifischen Sprache spezifiziert werden.

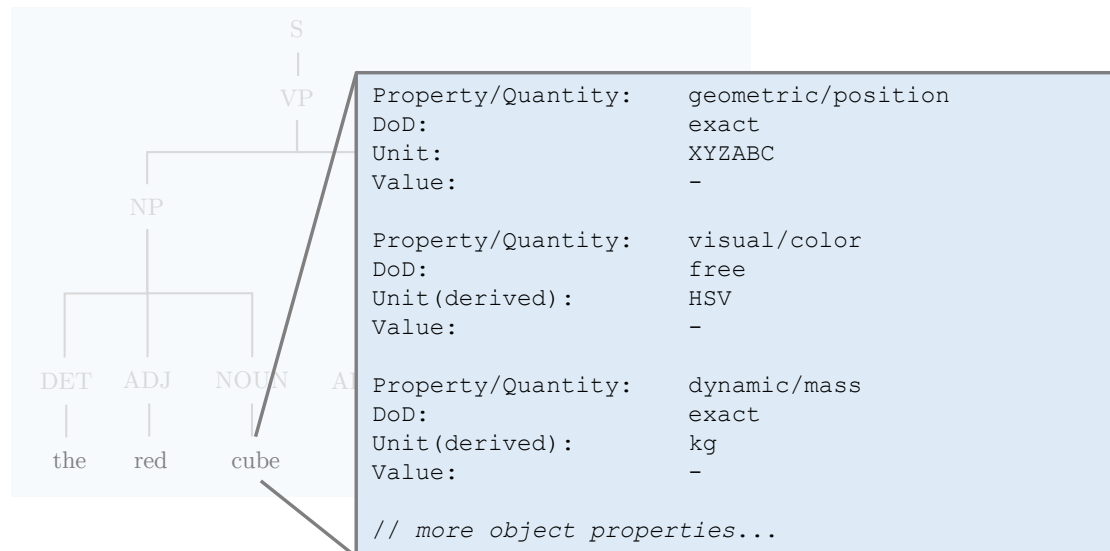


Abbildung 5.3: Ausschnitt der Annotation des Symbols `cube` mit Information aus dem physikalischen Wörterbuch.

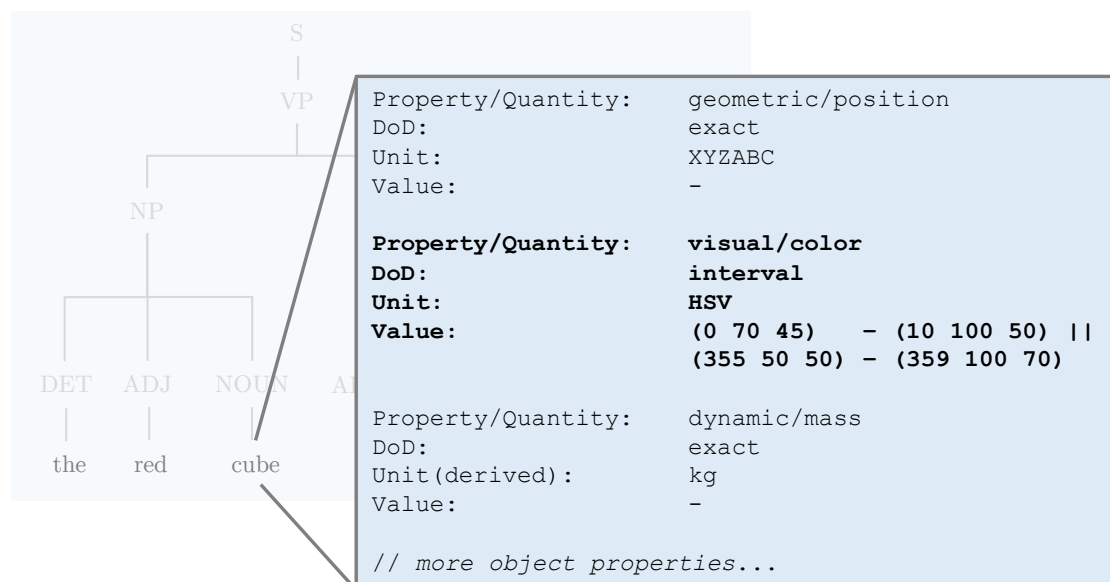


Abbildung 5.4: Einschränkung der physikalischen Eigenschaftswerte des Symbols `cube` aufgrund des im Rahmen der symbolischen Anweisung vorhandenen Parametersymbols `red`.

Im letzten Schritt des Algorithmus (siehe Zeile 15, Algorithmus 5.1) werden vorhandene Abhängigkeiten aufgelöst. Neben den Abhängigkeiten zwischen den im Rahmen der einzelnen Parameterphrasen spezifizierten Objekte und den Ein- und Ausgangsgrößen des auszuführenden Verbalisierten Physikalischen Effekts sind in der exemplarischen Anweisung auch beispielsweise Abhängigkeiten zwischen den Symbolen **the** und **cube** oder **red** und **cube** vorhanden. Betrachtet man die im Rahmen des physikalischen Wörterbuchs hinterlegte Information zu einem entsprechenden Symbol (siehe Abbildung 4.4), so ist dort hinterlegt, dass sowohl die Symbole **the** als auch **red** im Rahmen einer Nominalphrase auftreten können (siehe Eintrag **Function**). Des Weiteren ist hinterlegt, dass diese Symbole eine physikalische Eigenschaft eines Objekts festlegen, beziehungsweise einschränken (siehe Einträge zu **Symbol Semantics Information**). Diese Einschränkungen sind dem referenzierten Nomen zuzuweisen. Im Beispiel wird somit dem Symbol **cube** eine Einschränkung im Bezug auf die Anzahl sowie der Farbe zugeordnet (siehe Abbildung 5.4).

5.3 Extraktion von subsymbolischer Information

Basierend auf dem annotierten Phrasenstrukturbaum wird als nächster Schritt die zur Ausführung der instruierten Anweisung benötigte subsymbolische Information aus der Umgebung extrahiert und das zur auszuführenden Aktion zugeordnete Aktionsprimitivnetz parametrisiert. Im Folgenden werden die hierzu durchgeführten Schritte zunächst allgemein beschrieben. Anschließend werden diese auf Basis einer Beschreibung im Pseudocode dargestellt und anhand eines Beispiels im Detail erläutert.

Als erster Schritt wird das zur spezifizierten Aktion zugehörige Aktionsprimitivnetz instantiiert, welches in der entsprechenden XML-basierten Aktionsbeschreibung im System hinterlegt ist. An dieser Stelle ist insbesondere die Information der zu bestimmenden subsymbolischen Parameter relevant, welche unter anderem den Namen und Typ der Parameter sowie die zur Extraktion relevante obligatorische Ergänzung umfasst (siehe Code 5.6).

Im nächsten Schritt werden die im annotierten Phrasenstrukturbaum beschriebenen Objekte mit real vorhandenen Objekten verknüpft. Hierzu wird eine Komponente benötigt, welche dem Robotersystem eine Repräsentation der Umgebung zur Verfügung stellt. Unter Nutzung des physikalischen Wörterbuchs, einer Objektdatenbank sowie einer geeigneten Extraktionskomponente (siehe Abbildung 4.3) werden bekannte Objekte erkannt und mit der entsprechenden symbolischen Beschreibung annotiert. Im Rahmen der prototypischen Umsetzung wird hierzu eine Weltrepräsentation verwendet, deren Komponenten auf Basis des ENACT-Frameworks [Werner16] realisiert sind. Die zur Umsetzung des Prototyps verwendeten Komponenten werden in Kapitel 6.1 beschrieben. Zur Zuordnung der im Phrasenstrukturbaum beschriebenen Objekte zu realen Objekten der Umgebung werden die entsprechenden physikalischen Eigenschaften geprüft und validiert. Die zu verwendenden Extraktionskomponenten sind im Rahmen des physikalischen Wörterbuchs beschrieben (siehe Abbildung 4.4) und im System hinterlegt.

Sind die im Rahmen des annotierten Phrasenstrukturbaumes beschriebenen Objekte mit den entsprechenden realen Objekten verknüpft, können die zur Ausführung benötigten, subsymbolischen Parameter des Aktionsprimitivnetzes bestimmt werden. Diese werden auf Basis der physikalischen Eigenschaften der identifizierten, realen Objekte bestimmt. In Abhängigkeit der im System verfügbaren Extraktionskomponenten kann ein entsprechender Parameter entweder direkt oder indirekt bestimmt werden (siehe Kapitel 3.3). Betrachtet man beispielsweise eine physikalische Größe wie die Position oder Masse eines Objekts, so ist hierfür typischerweise eine Extraktionskomponente im System vorhanden, welche diese direkt aus der Umgebung extrahiert. Betrachtet man hingegen eine physikalische Größe wie die potentielle Energie, so wird diese typischerweise indirekt basierend auf einer im System hinterlegten physikalischen Gesetzmäßigkeit bestimmt. Diese setzt sich aus physikalischen Größen zusammen, für welche Extraktionskomponenten zur direkten Bestimmung im System vorhanden sind, beispielsweise die bereits genannten Komponenten zur Bestimmung der Position oder Masse eines Objekts. Können alle benötigten physikalischen Größen bestimmt werden, kann abschließend das zur instruierten Aktion zugeordnete Aktionsprimitivnetz parametrisiert und zur Ausführung an das Robotersystem weitergegeben werden.

Eingabe: annotierter Phrasenstrukturbaum O

Ausgabe: parametrisiertes Aktionsprimitivnetz MPN

```

1  function extract_subsymbolic_information(O)
2      MPN = load_mpn(O)
3      if (MPN != null)
4          world_objects = update_world()
5          foreach object in O
6              object.ID = map_objects(object, world_objects)
7              if (object.ID == null)
8                  return null                                /* ERROR_object_not_found */
9              end
10             foreach parameter in MPN
11                 foreach property in parameter
12                     properties = extract_property(property, object.ID)
13                 end
14                 if (properties != null)
15                     parameter = calculate_parameter(properties)
16                 end
17             else
18                 return null                                /* ERROR_load_mpn */
19             end
20             return MPN
21 end function

```

Algorithmus 5.2: Verfahren zur Extraktion von subsymbolischer Information sowie zur Parametrisierung des auszuführenden Aktionsprimitivnetzes.

Basierend auf der allgemeinen Beschreibung der durchzuführenden Schritte kann ein Algorithmus zur Extraktion der subsymbolischen Information und der Parametrisierung des auszuführenden Aktionsprimitivnetzes spezifiziert werden (siehe Algorithmus 5.2). Die Eingabe des Algorithmus stellt der annotierte Phrasenstrukturbaum dar, welchen das im Rahmen von Kapitel 5.2 beschriebene Verfahren liefert. Die Ausgabe stellt ein parametrisiertes Aktionsprimitivnetz dar, welches die durch das Robotersystem auszuführende Aktion beschreibt.

Im Folgenden werden die einzelnen Schritte des Algorithmus anhand einer exemplarischen Anweisung erläutert. Hierzu wird die im Rahmen des letzten Kapitels verwendete Instruktion

shove *the red cube* *towards a blue cylinder*

weiter genutzt. Den Ausgangspunkt stellt somit der annotierte Phrasenstrukturbaum dar, welcher gemäß des Verfahrens nach Algorithmus 5.1 erzeugt wird und in diesem zu jedem Symbol die im Rahmen des physikalischen Wörterbuchs vorhandene Information annotiert ist sowie entsprechende Referenzen (vgl. Abbildung 5.4) aufgelöst sind.

Der erste Schritt des Verfahrens stellt das Laden des zur Aktion zugeordneten Aktionsprimivnetzes dar, welches in einer parametrisierbaren Form im Rahmen der XML-basierten Aufgabenbeschreibung hinterlegt ist und die Semantik der auszuführenden Aktion beschreibt. Der allgemeine Teil der zum Schlüsselwort **shove** verwendeten XML-basierten Aktionsbeschreibung ist in Code 5.6 dargestellt, welcher bereits die beiden zu bestimmenden subsymbolischen Parameter **forceParam** beziehungsweise **lengthParam** beschreibt.

```
<!-- Specification of the manipulation primitive net -->
<verbbot:MPN>
  <!-- ... Manipulation primitives for approach phrase -->
  <verbbot:MP id="0"> </verbbot:MP>
  <!-- ... Manipulation primitives for effect execution phase -->
  <verbbot:MP id="1">
    <verbbot:HM>
      <verbbot:TASK_FRAME>
        <!-- ... Definition of the task frame -->
      </verbbot:TASK_FRAME>
      <verbbot:SELECTION>
        <verbbot:VALUE id="0" control="FORCE" ref="REF">
          forceParam
        </verbbot:VALUE>
        <!-- ... Definition for other DOFs -->
      </verbbot:SELECTION>
    </verbbot:HM>
    <verbbot:TOOL_COMMAND>
      <verbbot:GRIPPER command="CLOSE"/>
    </verbbot:TOOL_COMMAND>
    <verbbot:CRITERIA>
      <verbbot:SC id="2" type="POSITION" dim="0" ref="REF" relation="GREATER">
        lengthParam
      </verbbot:SC>
      <!-- ... Definition of other stopping criteria -->
    </verbbot:CRITERIA>
  </verbbot:MP>
  <!-- ... Manipulation primitives for release phase -->
  <verbbot:MP id="2"> </verbbot:MP>
  <!-- ... Definition of start primitives, stop primitives, edges of the MPN -->
</verbbot:MPN>
```

Code 5.7: Ausschnitt der XML-Aktionsdefinition zur Spezifikation des zur Aktion **shove** zugeordneten Aktionsprimivnetzes.

Der weitere Teil beschreibt das auszuführende Aktionsprimitivnetz sowie die Verwendung der Parameter (siehe Code 5.7). Zur besseren Übersicht ist hier ausschließlich eines der Aktionsprimitive der Manipulationsphase genauer spezifiziert, welches die Ausführung des Effekts beschreibt und beide benötigten Parameter erfordert. Der Eingangsparameter des Effekts `forceParam` wird als Sollwert der kraftgeregelten Bewegung eines Freiheitsgrads verwendet. Diese Bewegung wird solange ausgeführt, bis eines der Abbruchkriterien erreicht wird. Eines der Abbruchkriterien beschreibt hierbei die Distanz, um welche das manipulierte Objekt geschoben werden soll. Hierfür wird der Parameter `lengthParam` zur Definition verwendet.

Im nächsten Schritt des Verfahrens (Zeile 4, Algorithmus 5.2) wird der Zustand der Roboterumgebung sensorisch erfasst. Dies umfasst unter anderem die Erkennung der in der Umgebung vorhandenen Objekte, welche unter Nutzung einer dem System zur Verfügung stehenden Komponente durchgeführt wird. Je nach verfügbarer Sensorik nutzt diese die entsprechende Information des Sensors um beispielsweise im Vorfeld hinterlegte Objektdeskriptoren in einer Szene zu erkennen und die dadurch erkannten Objekte mit zugeordneten Bezeichnern zu annotieren. Eine exemplarische Szene sowie die auf Basis von im System vorhandener Information erkannter und annotierter Objekte ist in Abbildung 5.5 dargestellt.

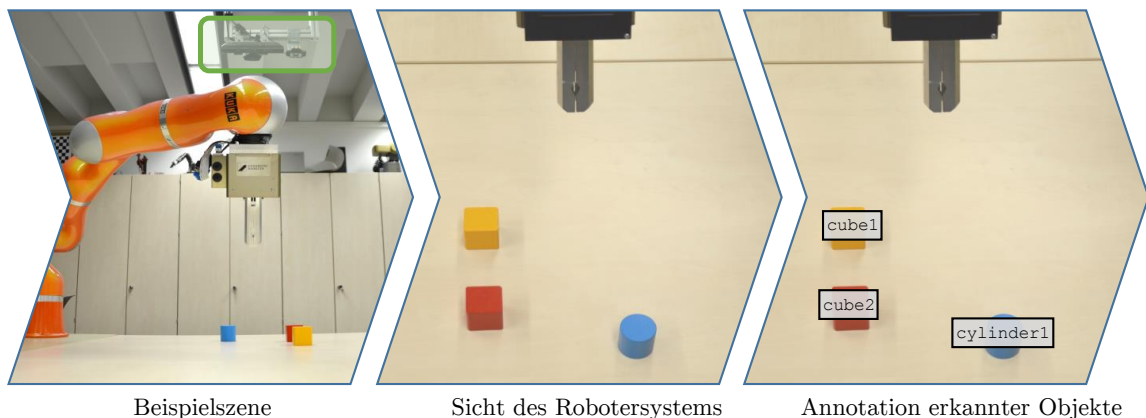


Abbildung 5.5: Erfassung und Annotation der in der Roboterumgebung vorhandenen Objekte. Im linken Bild ist eine Übersicht der Szene inklusive des zur Erfassung der Objekte verwendeten optischen Sensors (siehe grüne Umrandung) abgebildet. Im mittleren Bild ist die Szene aus Sicht des Robotersystems dargestellt. Im rechten Bild sind die auf Basis der im System verfügbaren Extraktionskomponente erkannten und annotierten Objekte illustriert.

Im nächsten Schritt des Verfahrens (Zeile 5-9, Algorithmus 5.2) werden die im Rahmen des annotierten Phrasenstrukturbaums spezifizierten Objekte den realen, erkannten Objekten zugeordnet. Hierzu werden die physikalischen Eigenschaften der im Phrasenstrukturbaum spezifizierten Objekte überprüft. Die benötigten Extraktionskomponenten sind im Rahmen des physikalischen Wörterbuchs beschrieben. Ein Verweis auf diese wird beim Auflösen der symbolischen Information (siehe Algorithmus 5.1) an die Symbole hinzugefügt. Somit können beispielsweise die zur Ausführung benötigten Objekte `cube` und `cylinder` erkannt werden, insbesondere kann das Objekt `cube` mit den entsprechenden physikalischen Eigenschaften erkannt werden (siehe Abbildung 5.6).

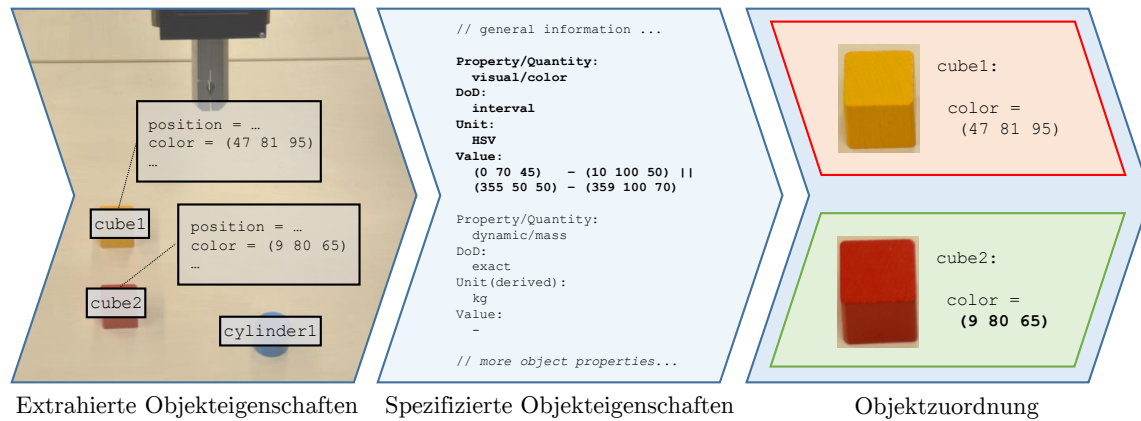


Abbildung 5.6: Zuordnung des realen Objekts **cube** zu dem im Rahmen des annotierten Phrasenstrukturbaums spezifizierten Objekts (siehe Abbildung 5.4).

Sind die zur Ausführung relevanten Objekte identifiziert, können abschließend die im Rahmen des Aktionsprimitivnetzes benötigten Parameter extrahiert, beziehungsweise berechnet werden (Zeile 10-15, Algorithmus 5.2). Im Beispiel sind dies zum einen der Parameter **forceParam**, welcher auf Basis der physikalischen Eigenschaften des Objekts berechnet wird, welches im Rahmen der Nominalphrase der instruierten Anweisung beschrieben ist. Zum anderen ist dies der Parameter **lengthParam**, welcher auf Basis der physikalischen Eigenschaften der beiden spezifizierten Objekte berechnet wird.

5.4 Zusammenfassung

Im Rahmen dieses Kapitels wurde die wissenschaftliche Fragestellung **F3** betrachtet, welche die Parametrisierung von Roboteraktionen basierend auf symbolisch instruierten Anweisungen sowie die Integration der vorgeschlagenen Konzepte und im System verfügbaren Extraktionskomponenten adressiert. Diese lautet wie folgt:

- F3** In welcher Form können die vorgeschlagenen Konzepte zur Beschreibung der Symbolsemantik gemeinsam verwendet werden um eine subsymbolische Parametrierung der auszuführenden Roboteraktionen zu ermöglichen?

Zur gemeinsamen und intuitiven Nutzung der im Rahmen von Kapitel 3 und Kapitel 4 beschriebenen Konzepte in Form einer symbolischen Benutzerschnittstelle wurde die Verwendung im Rahmen einer domänenspezifischen Sprache vorgeschlagen. Diese nutzt die zu einem auszuführenden Verbalisierten Physikalischen Effekt zugeordneten Verben als Schlüsselworte und die entsprechenden obligatorischen Ergänzungen in einer geeigneten Ausdrucksform als Parameterphrasen. Zur Definition von validen Anweisungen werden die ausführbaren Aktionen auf Basis einer XML-basierten Beschreibung im System hinterlegt, welche gemäß eines XML-Schemas spezifiziert werden. Diese Form der Aktionsbeschreibung wird unabhängig vom Robotersystem spezifiziert, wodurch diese flexibel für verschiedene Robotersysteme genutzt werden kann.

Zur Parametrisierung der auszuführenden Roboteraktionen basierend auf den symbolisch instruierten Anweisungen wurden zwei Verfahren vorgestellt, welche zum einen das Auflösen von symbolischer Information und zum anderen die Extraktion von subsymbolischer Information als auch die Parametrisierung des auszuführenden Aktionsprimitivnetz beschreiben. Das Verfahren zum Auflösen der symbolischen Information nutzt die im Robotersystem hinterlegte Symbolinformation, welche im Rahmen des physikalischen Wörterbuchs und der XML-basierten Aktionsbeschreibung zur Verfügung steht. Das Verfahren zur Extraktion und zur Parametrisierung des auszuführenden Aktionsprimitivnetzes nutzt insbesondere die zur Verfügung stehenden Extraktionskomponenten zur Bestimmung der physikalischen Eigenschaften von realen Objekten der Roboterumgebung.

Abschließend sind die Zusammenhänge, welche im Rahmen dieses Kapitels erarbeitet und beschrieben wurden, in schematischer Form in Abbildung 5.7 dargestellt.

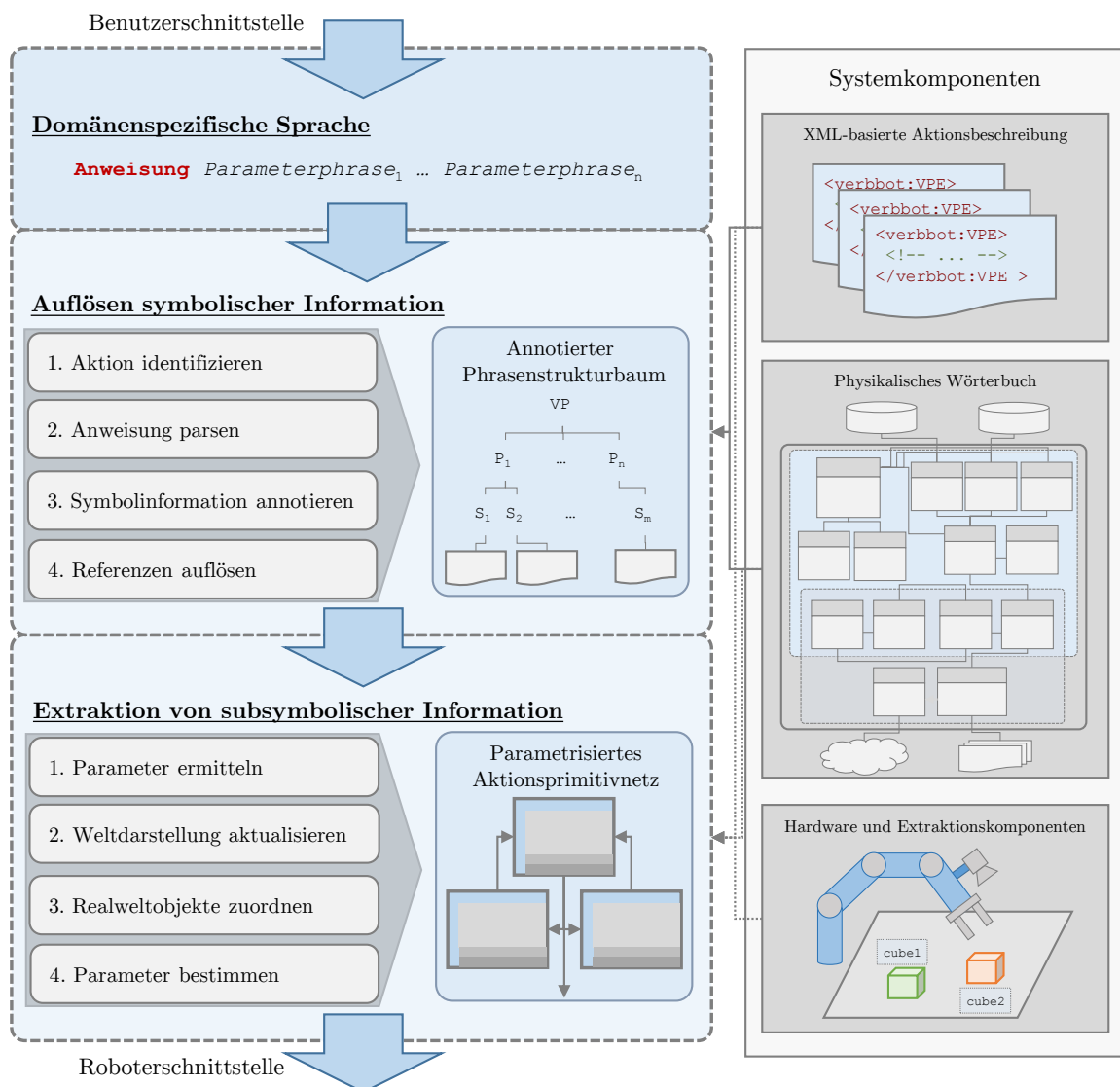


Abbildung 5.7: Schematische Zusammenhänge der im Rahmen dieses Kapitels beschriebenen Verfahren.

6 Evaluation

Die in Kapitel 3, 4 und 5 vorgestellten Konzepte und Verfahren stellen die benötigten Komponenten zur Verfügung, welche die in der Aufgabenstellung (siehe Kapitel 1.2) spezifizierte Transformationsschicht zwischen einer symbolischen Benutzerschnittstelle und einer subsymbolischen Roboterschnittstelle beschreiben. Diese sind im Rahmen eines prototypischen Gesamtsystems umgesetzt, welches im Folgenden beschrieben und als Basis der weiteren Evaluation der Konzepte und Verfahren genutzt wird.

Zunächst werden die einzelnen Hard- und Softwarekomponenten beschrieben, welche zur prototypischen Umsetzung genutzt werden (siehe Kapitel 6.1). Dies umfasst eine Darstellung der verwendeten Robotersysteme inklusive der zur Verfügung stehenden Sensorik, der Weltrepräsentation auf Basis des ENACT-Frameworks [Werner16] sowie der grafischen Benutzerschnittstelle, welche zur Eingabe von auszuführenden Anweisungen und zur Visualisierung der einzelnen Verarbeitungsschritte genutzt wird.

Ein Merkmal der Definition von Roboteraktionen unter Nutzung der Verbalisierten Physikalischen Effekte stellt die Aufgabenraum-basierte Spezifikation der auszuführenden Aktionen dar, welche in Form von Aktionsprimitivnetzen beschrieben werden. Diese besitzen die Eigenschaft, eine vom ausführenden Robotersystem unabhängige Beschreibung der Aktionsdefinition zu erlauben (siehe Kapitel 2.2). Daher kann die Aktionsdefinition unter Nutzung der Verbalisierten Physikalischen Effekte für verschiedene Roboterkonfigurationen genutzt werden, welche sich beispielsweise aus verschiedenen Hard- und Softwarekomponenten zusammensetzen kann. Im Rahmen von Kapitel 6.2 wird dieses Merkmal unter Nutzung von unterschiedlichen Robotersystemen evaluiert.

Neben der flexiblen Verwendung der vorgeschlagenen Konzepte steht die intuitive Nutzung im Rahmen einer symbolischen Nutzerschnittstelle im Fokus dieser Arbeit. In der Aufgabenstellung (siehe Kapitel 1.2) ist hierzu die wissenschaftliche Fragestellung **F4** formuliert. Diese lautet wie folgt:

F4 In wie weit kann basierend auf den vorgeschlagenen Konzepten ein intuitiv zu verwendendes Gesamtsystem umgesetzt werden?

Zur Umsetzung einer intuitiven, symbolischen Benutzerschnittstelle ist im Rahmen von Kapitel 5.1 die Nutzung einer domänenspezifischen Sprache vorgeschlagen, welche die zu einem verbalisierten physikalischen Effekt zugeordneten Ausdrücke als Schlüsselwörter und deren obligatorischen Ergänzungen in geeigneten Ausdrucksformen als Parameterphrasen verwendet.

In wie weit diese Form als intuitive, symbolische Nutzerschnittstelle geeignet ist, wird in Kapitel 6.3 untersucht. Hierzu werden im Rahmen einer Benutzerstudie empirische Daten erhoben, welche zum einen die Eingaben in Form der domänenspezifischen Sprache und zum anderen die Ausgaben in Form der auszuführenden Aktionsprimitivnetze evaluieren. Abschließend werden die im Rahmen der Evaluation gewonnenen Erkenntnisse und Ergebnisse zusammengefasst und diskutiert (siehe Kapitel 6.4).

6.1 Prototypische Umsetzung

Im Folgenden werden die zur prototypischen Umsetzung genutzten Hard- und Softwarekomponenten erläutert. Dies umfasst insbesondere die Darstellung der Komponenten, welche für beide Systeme gemeinsam genutzt werden, sowie die notwendigen Anpassungen, falls diese auf Basis der verschiedenen Sensorik der Robotersysteme nicht gemeinsam genutzt und individuell umgesetzt werden müssen. Die Ansteuerung der Hardwarekomponenten sowie die Implementierung der Softwarekomponenten ist im Allgemeinen in C++ realisiert, vorhandene Abhängigkeiten zu entsprechenden Bibliotheken werden bei der Beschreibung einzelner Komponenten genannt.

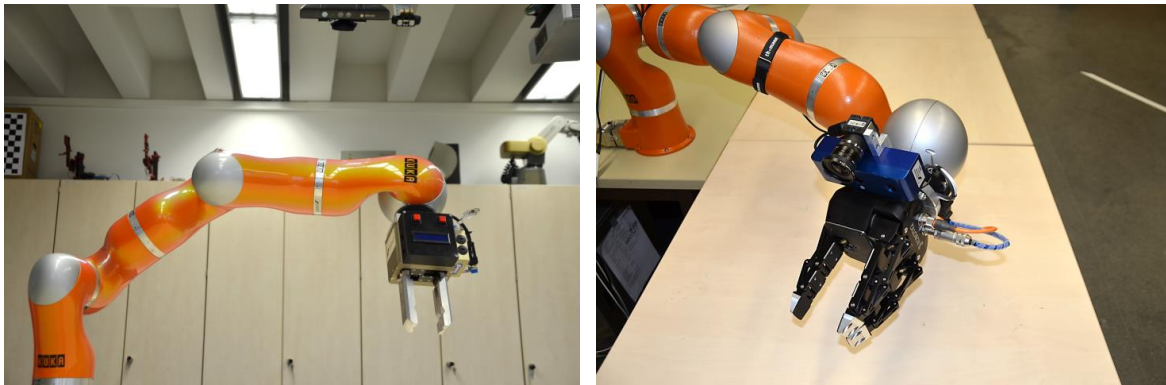


Abbildung 6.1: Illustration der zur prototypischen Umsetzung verwendeten Robotersysteme. Das System aus der linken Abbildung setzt sich aus einem KUKA LBR4, einem 2-Finger-Parallelgreifer PG70 von Schunk, sowie einer Microsoft Kinect zusammen. Das System aus der rechten Abbildung setzt sich aus einem KUKA LBR4+, einem adaptiven 3-Finger-Robotergrifer von Robotiq, einem Ensensio N10 Tiefensensor und einer IDS UI-1220SE Industriekamera zusammen.

Im Bereich der Hardwarekomponenten werden zwei unterschiedliche Robotersysteme verwendet, welche in Abbildung 6.1 dargestellt sind. Der allgemeine Aufbau der Systeme ist identisch, so setzt sich jedes aus einem stationären Roboterarm, welcher über interne Kraft-/Momentensensorik verfügt, einem Greifer zur Objektmanipulation sowie optischer Sensorik zur Erfassung der Umgebung zusammen.

Im Bereich der Softwarekomponenten stehen für beide Robotersysteme einige gemeinsam genutzte Komponenten zur Verfügung, beispielsweise den allgemeinen Komponenten zur Verarbeitung von Eingaben gemäß der domänenspezifischen Sprache (siehe Algorithmus 5.1, 5.2) die XML-basierte Aktionsbeschreibung, die Weltrepräsentation sowie die grafische Benutzerschnittstelle, welche im Folgenden beschrieben werden.

Eine gemeinsam verwendete Komponente stellt die XML-basierte Aktionsbeschreibung dar, welche die auszuführenden Aktionen auf Basis der hinterlegten Aktionsprimitivnetze beschreiben. Diese sind im System hinterlegt und umfassen die Definition einer Reihe von Aktionen wie beispielsweise `lift`, `move`, `rotate`, `shove`, `sink` oder `touch`, welche als Schlüsselworte der domänenspezifischen Sprache (siehe Kapitel 5.1) zur Verfügung stehen und zur Instruktion von symbolischen Anweisungen genutzt werden können.

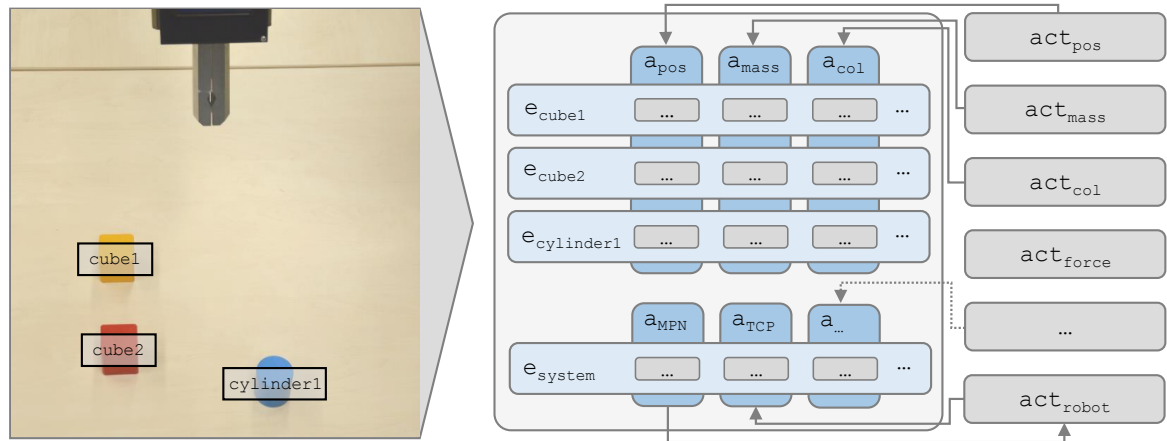


Abbildung 6.2: Darstellung einer Beispielszene (links) und zugehörige Komponenten der Weltrepräsentation basierend auf dem ENACT Framework (rechts). Für jedes erkannte Objekt der Szene ist eine Entität mit einer Reihe von Aspekten vorhanden, welche über entsprechende Aktoren spezifiziert werden. Zudem ist typischerweise eine Entität vorhanden, welche das Robotersystem beschreibt.

Zur Parametrisierung der auszuführenden Aktion wird eine Weltrepräsentation benötigt, welche auf Basis des ENACT-Frameworks umgesetzt ist. Dieses beschreibt ein Rahmenwerk für den Entwurf von modularen Robotikanwendungen, welche sich aus statischen und dynamischen Komponenten zusammensetzen. Die statischen Komponenten beschreiben hierbei die in der Roboterumgebung vorhandenen Objekte, deren Eigenschaften durch dynamische Komponenten verändert werden können (siehe Abbildung 6.2).

Die statischen Komponenten der Weltrepräsentation stellen Entitäten und Aspekte dar, wobei Entitäten reale Objekte der Umgebung und Aspekte die Eigenschaften von Entitäten repräsentieren. Im Kontext dieser Arbeit entsprechen die Aspekte insbesondere den physikalischen Eigenschaften eines Objekts wie beispielsweise der Position, Masse oder Farbe.

Die dynamischen Komponenten stellen die Aktoren dar, welche die Aspekte von Entitäten modifizieren können. Im Kontext dieser Arbeit entsprechen diese somit insbesondere den Extraktionskomponenten, welche die gesuchten Objekteigenschaften aus der Umgebung extrahieren und im Rahmen des physikalischen Wörterbuchs beschrieben sind.

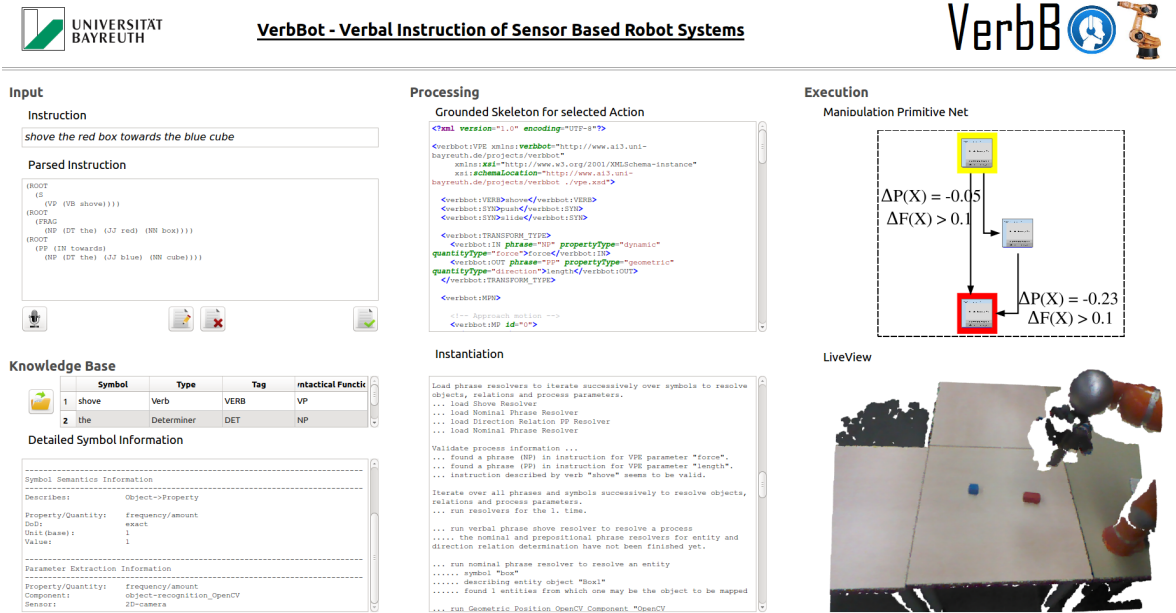


Abbildung 6.3: Grafische Benutzeroberfläche zur Eingabe von auszuführenden Anweisungen, zum Laden der Wissensbasis und zur Visualisierung der einzelnen Verarbeitungsschritte.

Die grafische Benutzerschnittstelle wird ebenfalls für beide Robotersysteme gemeinsam verwendet. Die im Rahmen der prototypischen Umsetzung genutzte Benutzerschnittstelle ist in Abbildung 6.3 abgebildet und auf Basis von QT [QT17] realisiert. Diese visualisiert relevante Informationen von der Eingabe einer auszuführenden Instruktion gemäß der domänenspezifischen Sprache bis hin zur Ausführung der Aktion basierend auf dem parametrisierten Aktionsprimitivnetz. Im Bereich der Eingabe steht Funktionalität zum Laden einer Wissensbasis zur Verfügung, welche die zur Spezifikation von auszuführenden Anweisungen verfügbaren Symbole beschreibt und sich aus der XML-basierten Aktionsbeschreibung und dem physikalischen Wörterbuch zusammensetzt. Des Weiteren steht Funktionalität zur Verfügung, um auf Basis der bekannten Symbole auszuführende Anweisungen gemäß der domänenspezifischen Sprache zu spezifizieren. Bei der Verarbeitung und Ausführung einer instruierten Anweisung werden zudem diverse Schritte visuell aufbereitet und angezeigt. Diese umfassen beispielsweise die Darstellung des resultierenden Phrasenstrukturbaums einer instruierten Anweisung, die Anzeige der im System hinterlegten Symbolinformation, die Darstellung der zu parametrisierenden XML-basierten Aktionsbeschreibung oder die Visualisierung der Aktionsausführung.

Neben den gemeinsam verwendeten Komponenten werden auch einige individuelle Komponenten benötigt. Dies ist im Allgemeinen der Fall, falls eine Komponente eine spezielle Sensorik erfordert. Im Rahmen der prototypischen Umsetzung betrifft dies zum einen die zur Extraktion von subsymbolischer Information genutzten Extraktionskomponenten, da beispielsweise verschiedene optische Sensoren oder Greifsysteme bei den beiden Robotersystemen verwendet werden. Zum anderen betrifft dies einen Teil des physikalischen Wörterbuchs, insbesondere den Teil, welcher die zur Extraktion verwendete Komponente und Sensorik spezifiziert.

Ein Beispiel für die notwendigen Anpassungen ist in Abbildung 6.4 dargestellt. Im Rahmen der prototypischen Umsetzung wird für jedes Robotersystem ein individuell angepasstes physikalisches Wörterbuch genutzt, welches die entsprechenden Sensoren und Extraktionskomponenten berücksichtigt. Dies ist in Form einer lokalen, relationalen *SQLite* [SQLite17] Datenbank im Robotersystem hinterlegt.

<pre> ----- General Symbol Information ----- Symbol: red Synonym: - Type: Adjective Tag: ADJ Function: ADJP,NP ----- Symbol Semantics Information ----- Describes: Object->Property DB: - ----- Property/Quantity: visual/color DoD: interval Unit: HSV Value: (0 70 45) - (10 100 50) (355 50 50) - (359 100 70) </pre>	<pre> ----- General Symbol Information ----- Symbol: red Synonym: - Type: Adjective Tag: ADJ Function: ADJP,NP ----- Symbol Semantics Information ----- Describes: Object->Property DB: - ----- Property/Quantity: visual/color DoD: interval Unit: HSV Value: (0 70 45) - (10 100 50) (355 50 50) - (359 100 70) </pre>
<pre> ----- Parameter Extraction Information ----- Property/Quantity: visual/color Component: color-recog_OCV Sensor: uEye </pre>	<pre> ----- Parameter Extraction Information ----- Property/Quantity: visual/color Component: color-recog_OCV Sensor: kinect </pre>

Abbildung 6.4: Beispiel für die Anpassung der im physikalischen Wörterbuch hinterlegten Symbolinformation zur Extraktion einer physikalischen Eigenschaft für verschiedene Robotersysteme. Die allgemeine Beschreibung der Symbolsemantik ist unabhängig vom Robotersystem spezifiziert und kann daher allgemein verwendet werden (grau). Die zur Extraktion verwendete Sensorik muss an entsprechender Stelle spezifiziert werden (blaue Umrandung).

6.2 Flexible Verwendung der Aktionsbeschreibung

Ein Merkmal des im Rahmen von Kapitel 3 eingeführten Konzepts der Verbalisierten Physikalischen Effekte stellt die Aufgabenraum-basierte Aktionsbeschreibung dar, welche eine vom Robotersystem unabhängige Aktionsspezifikation erlaubt. Im Rahmen dieses Kapitels wird zunächst untersucht, welche Eigenschaften unabhängig spezifiziert, beziehungsweise bei der Ausführung einer Anweisung modifiziert werden können, ohne eine Veränderung der gegebenen Anweisung zu erfordern. Darauf aufbauend ist zu untersuchen, in welcher Form eine entsprechende Veränderung in der flexiblen Aktionsspezifikation berücksichtigt, beziehungsweise kompensiert werden kann. Anhand der identifizierten Eigenschaften werden entsprechende Anwendungsszenarien zur Evaluation abgeleitet, ausgeführt, ausgewertet und diskutiert.

Eine der modifizierbaren Eigenschaften stellt das auszuführende Robotersystem dar. Besitzt ein System die zur Ausführung benötigten Sensoren, beziehungsweise Extraktionskomponenten sowie eine Schnittstelle zur Verarbeitung von Aktionsprimitivnetzen, so ist dieses prinzipiell in der Lage, die durch einen Verbalisierten Physikalischen Effekt spezifizierte Aktion auszuführen. Diese Voraussetzungen sind für die im Rahmen der prototypischen Umsetzung verwendeten Robotersysteme (siehe Abbildung 6.1) gegeben. Daher ist anhand von ausgewählten Anwendungsszenarien zu prüfen, in wie weit diese Systeme eine gegebene Anweisung unter Nutzung der identischen, im jeweiligen System hinterlegten Aktionsbeschreibung ausführen können.



Abbildung 6.5: Beispiele für Variationen der Roboterumgebung, welche auf Basis der Spezifikation von sensorbasierten Roboteraktionen kompensiert werden können.

Eine weitere modifizierbare Eigenschaft stellt die Umwelt, beziehungsweise die Umgebung des Robotersystems dar. Da die im Rahmen der Aktionsbeschreibung spezifizierten Roboterbewegungen im Allgemeinen in Form von sensorbasierten Bewegungen hinterlegt sind, erlaubt dies prinzipiell die Ausführung einer Anweisung in verschiedenen Umgebungen wie beispielsweise in Abbildung 6.5 dargestellt. Daher ist ebenfalls zu prüfen, in wie weit ein System eine instruierte Anweisung unter Nutzung der hinterlegten Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten für verschiedene Roboterumgebungen ausführen kann.

Die Modifikation der Umwelt, beziehungsweise der Roboterumgebung umfasst zudem die Variation der Eigenschaften von Objekten, welche im Rahmen einer auszuführenden Aktion manipuliert werden. Bei Verwendung in Form der domänenspezifischen Sprache werden diese aus den entsprechenden Parameterphrasen extrahiert, anhand der spezifizierten physikalischen Eigenschaften zugeordnet und dienen als Basis zur Parametrisierung des Aktionsprimitivnetzes (siehe Kapitel 5). Besitzt ein System die zur Parametrisierung benötigten Extraktionskomponenten, so ist dies prinzipiell in der Lage, eine instruierte Anweisung unter Nutzung veränderter Objekteigenschaften (siehe Abbildung 6.6) auszuführen, was ebenfalls in Form von ausgewählten Anwendungsszenarios zu prüfen ist.

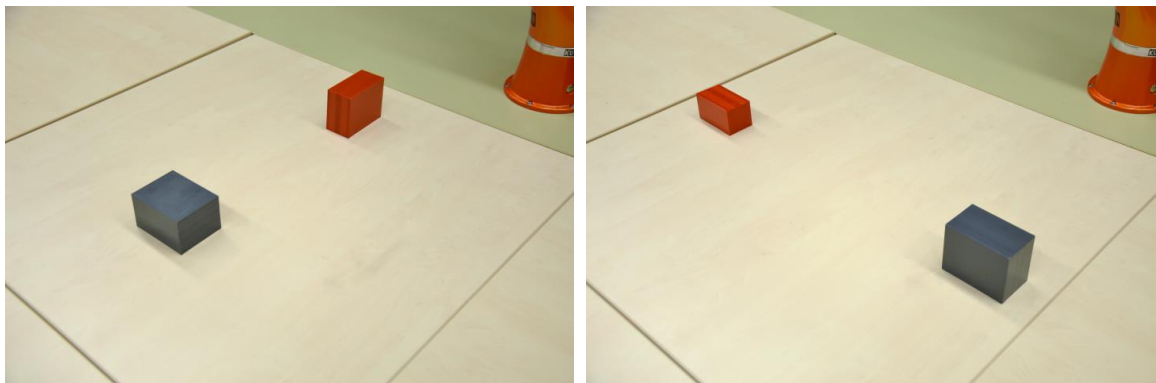


Abbildung 6.6: Beispiele für Variationen der zu manipulierenden Objekte, beziehungsweise Objekteigenschaften, welche auf Basis der Extraktionskomponenten kompensiert werden können.

Neben den genannten Eigenschaften lassen sich weitere Merkmale wie beispielsweise die Kompensation von Abweichungen oder Fehlern identifizieren. Diese werden im Rahmen dieser Arbeit jedoch nicht betrachtet (siehe Kapitel 1.3) und werden beispielsweise in [Orendt15] untersucht.

Zusammenfassend lassen sich somit drei Merkmale festhalten, welche im Rahmen der Ausführung einer instruierten Anweisung variiert werden, um die Flexibilität der Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten zu evaluieren. Diese lauten wie folgt:

- Variation des Robotersystems
- Variation der Roboterumgebung
- Variation der Objekteigenschaften

Als Varianten des Robotersystems werden die Systeme verwendet, welche in Abbildung 6.1 dargestellt sind. Die Varianten der Roboterumgebung sowie der Objekteigenschaften sind in Abbildung 6.5, beziehungsweise Abbildung 6.6 ersichtlich. Aufgrund der gewählten Varianten resultieren insgesamt 2^3 Möglichkeiten der Ausführung einer identischen Anweisung. Diese werden im Folgenden für ein Anwendungsszenario im Detail beschrieben.

Zur Evaluation wird exemplarisch die Anweisung

shove *the red cuboid* *towards the gray cuboid*

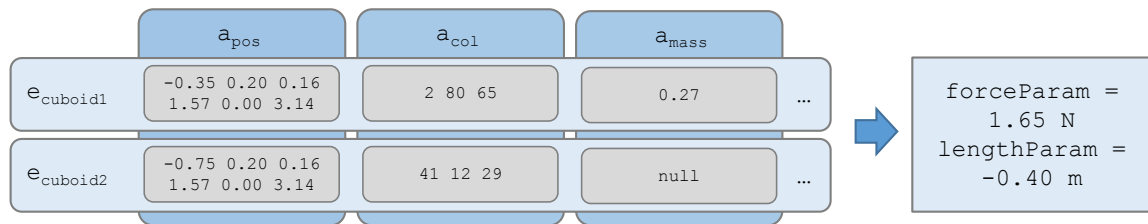
verwendet, welche gemäß der domänenspezifischen Sprache instruiert wird. Die Aktion erfordert eine sensorbasierte Ausführung, welche zur Parametrisierung eine Reihe von geometrischen und dynamischen Objekteigenschaften benötigt (siehe Kapitel 5.3). Die zu extrahierenden physikalischen Parameter stellen zum einen die Lage, Farbe und Masse des Objekts dar, welches durch die Parameterphrase **the red cuboid** spezifiziert wird. Zum anderen wird die Lage und Farbe des Objekts benötigt, welches im Rahmen der Parameterphrase **towards the gray cuboid** spezifiziert wird. Diese werden durch die entsprechenden Extraktionskomponenten bestimmt, welche im Rahmen des physikalischen Wörterbuchs der verwendeten Robotersysteme hinterlegt sind (siehe Kapitel 6.1). Weitere benötigte Größen wie beispielsweise die Erdbeschleunigung oder entsprechende Reibungskoeffizienten sind in Form von Standardwerten in den Systemen hinterlegt.

Bevor die Ausführungen der einzelnen Varianten analysiert werden, werden zunächst die relevanten Stammdaten der Ausführungsvarianten dargestellt. Diese werden im Folgenden genutzt, um die aufgrund der automatischen Extraktion vorhandenen Abweichungen der zur Ausführung benötigten Parameterwerte zu vergleichen und zu bewerten. Die relevanten Stammdaten sind im Wesentlichen durch die Varianten des Objektsatzes festgelegt, welcher die Objekteigenschaften der zu manipulierenden Objekte beschreibt.

Wie im Rahmen von Kapitel 6.1 beschrieben, sind die Objekte der Roboterumgebung in Form von Entitäten und Aspekten gemäß dem ENACT-Framework im System repräsentiert (vgl. Abbildung 6.2). Die Aspekte der entsprechenden Objekte werden zur Berechnung der Aktionsparameter **forceParam** beziehungsweise **lengthParam** verwendet (siehe Code 5.7). Die zur Ausführung der einzelnen Umweltvarianten verwendeten Objektparameter sind in Abbildung 6.7 dargestellt.

Im Folgenden werden die Anweisungsausführung sowie die extrahierten Parameter der einzelnen Varianten beschrieben. Zur Bezeichnung der einzelnen Varianten werden die in Abbildung 6.1 dargestellten Robotersysteme als **System1** (linke Abbildung) und **System2** (rechte Abbildung) bezeichnet. Die im Rahmen von Abbildung 6.5 gezeigten Varianten der Roboterumgebung werden als **Umgebung1** (links) und **Umgebung2** (rechts) bezeichnet. Analog werden die in Abbildung 6.6 illustrierten Varianten als **Objektsatz1** (links) und **Objektsatz2** (rechts) bezeichnet.

Stammdaten für Objektsatz1



Stammdaten für Objektsatz2

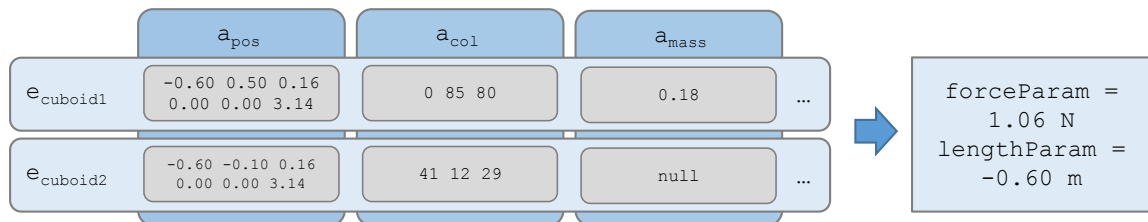


Abbildung 6.7: Stammdaten der im Rahmen der Evaluation verwendeten Objektsätze.

In den Abbildungen 6.8 - 6.15 sind die extrahierten Parameter sowie die resultierende Ausführung der exemplarischen Anweisungen für die acht Umgebungsvarianten dargestellt.

Im oberen Teil der Abbildungen sind jeweils die extrahierten Parameterwerte dargestellt, welche bei der Anweisungsverarbeitung auf Basis der im System hinterlegten Extraktionskomponenten bestimmt und berechnet werden (vgl. Abbildung 5.7). Die dargestellten Parameterwerte entsprechen dem Mittelwert, welcher auf Basis einer mehrfachen Aktionsausführung ermittelt wird. Die Abweichungen der Parameterbestimmung werden im Anschluss diskutiert.

Im unteren Teil einer jeweiligen Abbildung sind verschiedene Zeitschritte der Aktionsausführung visualisiert, wobei das erste Bild die Ausgangssituation und das sechste Bild den Endzustand der Ausführung kennzeichnet.

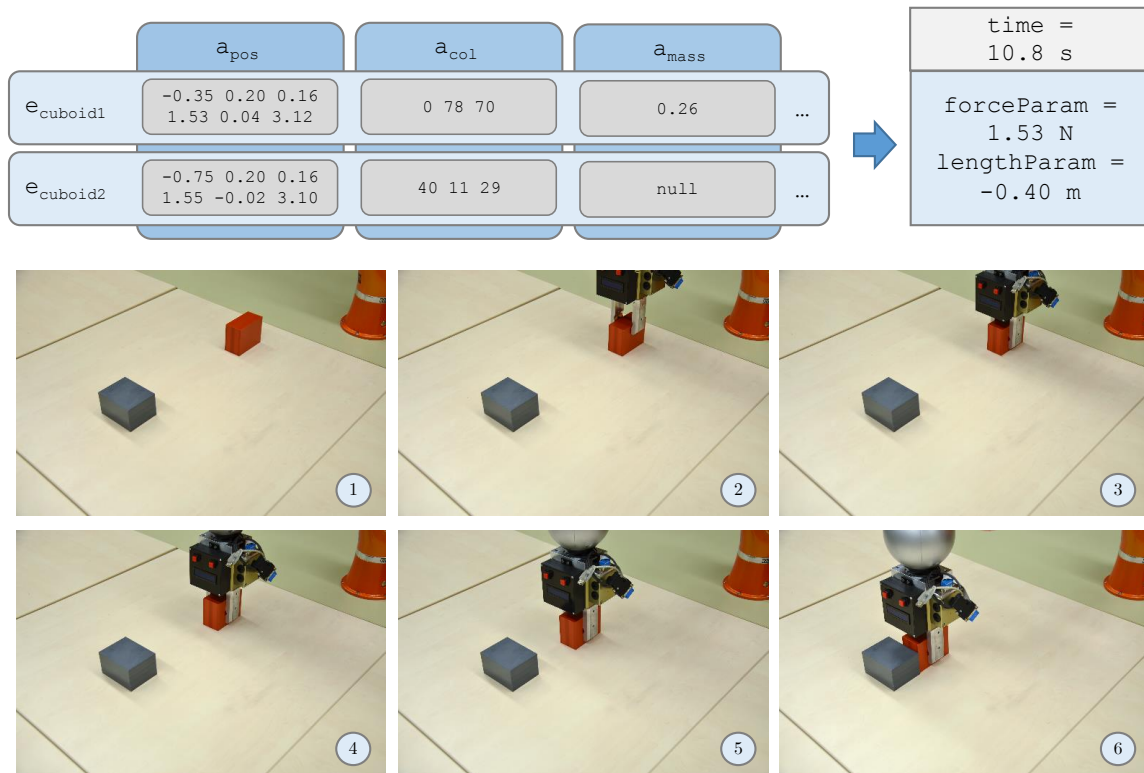


Abbildung 6.8: Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System1, Umgebung1 und Objektsatz1.

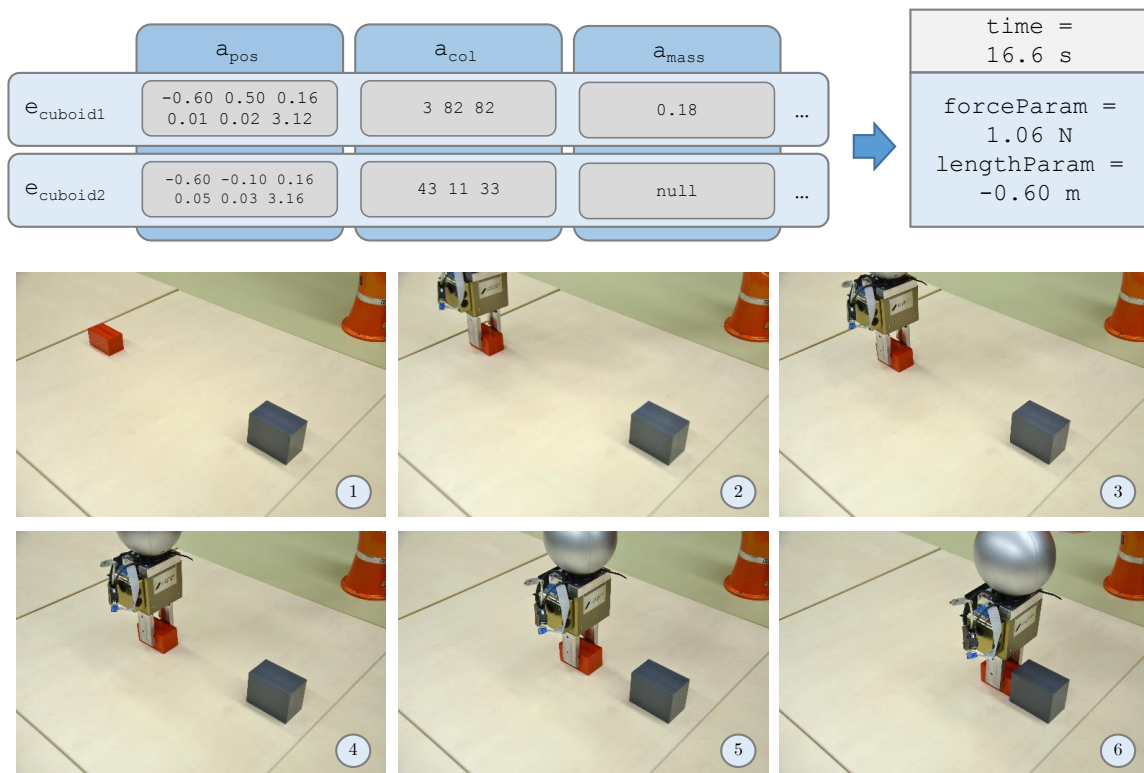


Abbildung 6.9: Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System1, Umgebung1 und Objektsatz2.

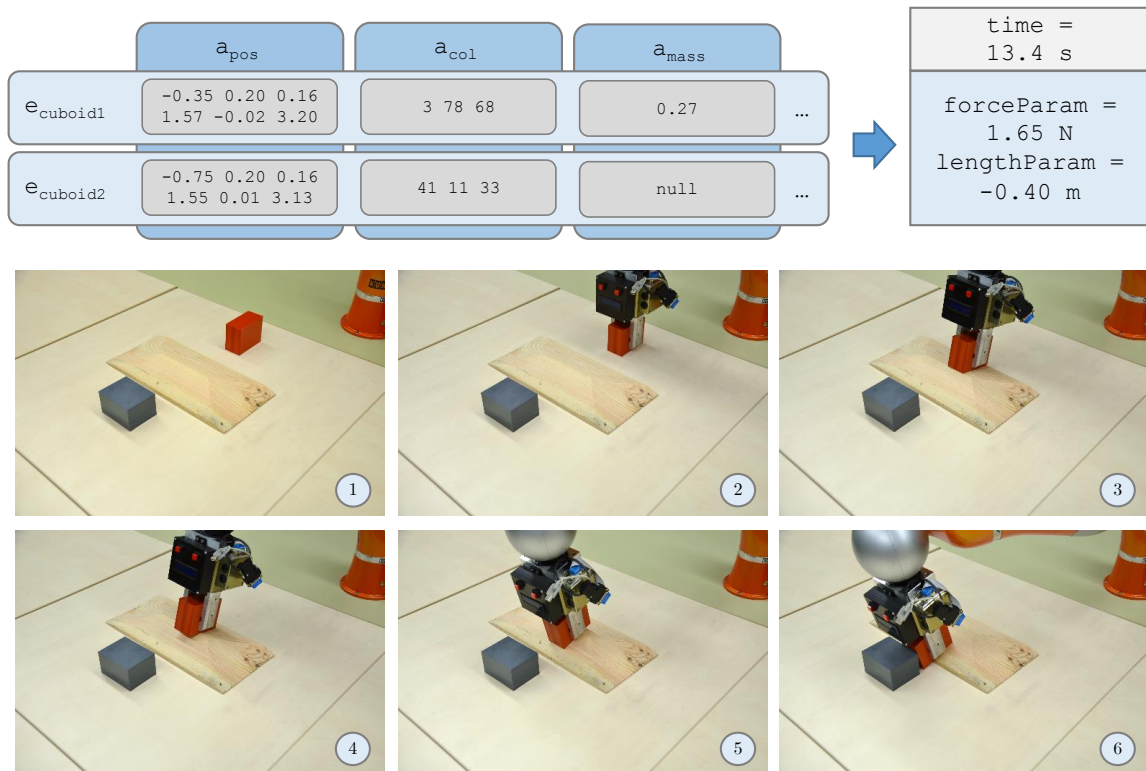


Abbildung 6.10: Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System1, Umgebung2 und Objektsatz1.

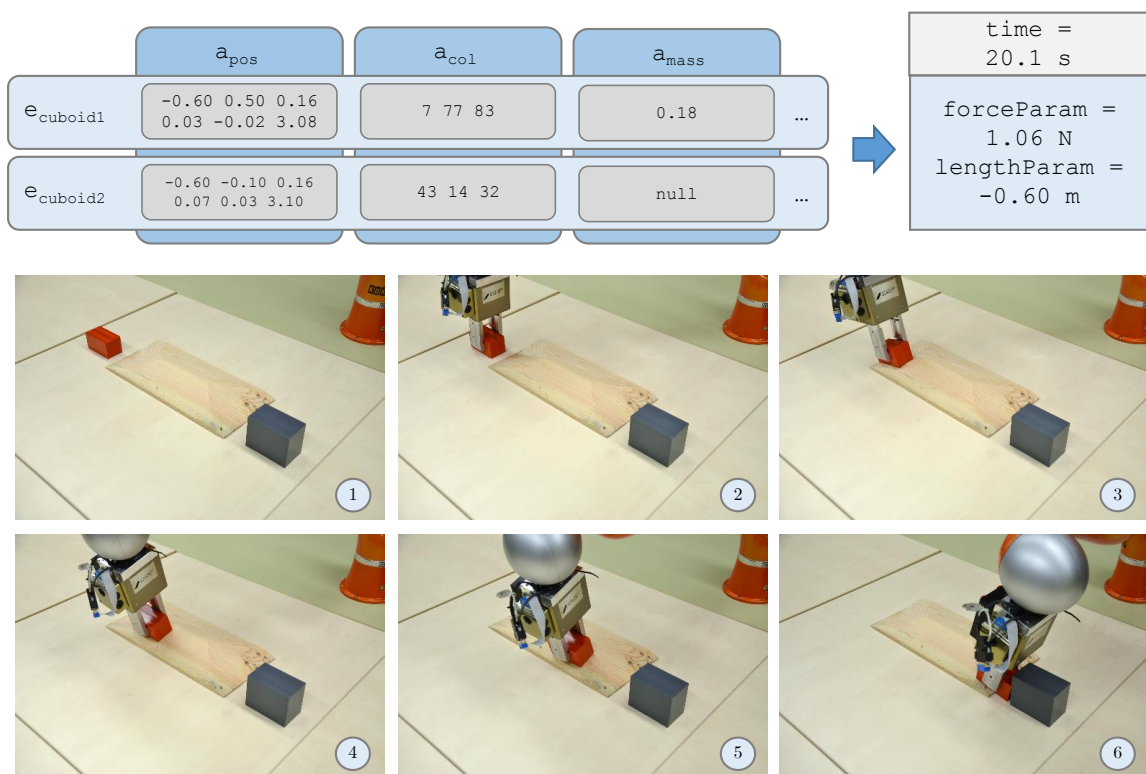


Abbildung 6.11: Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System1, Umgebung2 und Objektsatz2.

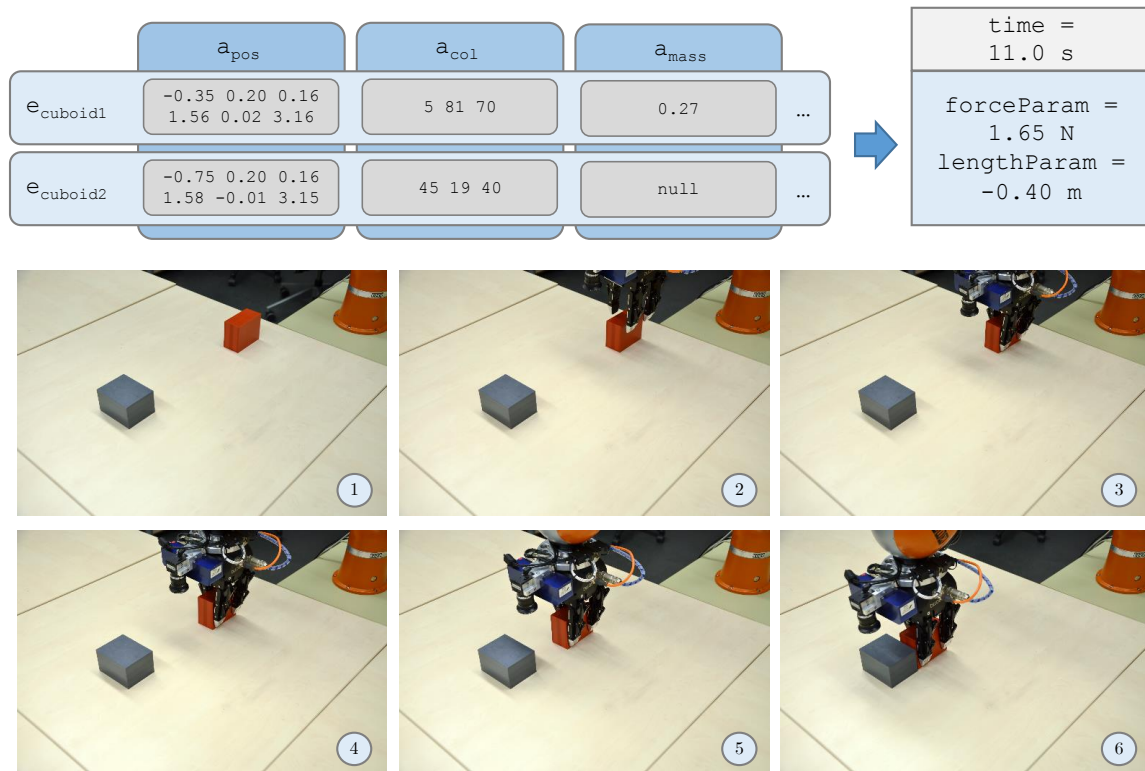


Abbildung 6.12: Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System2, Umgebung1 und Objektsatz1.

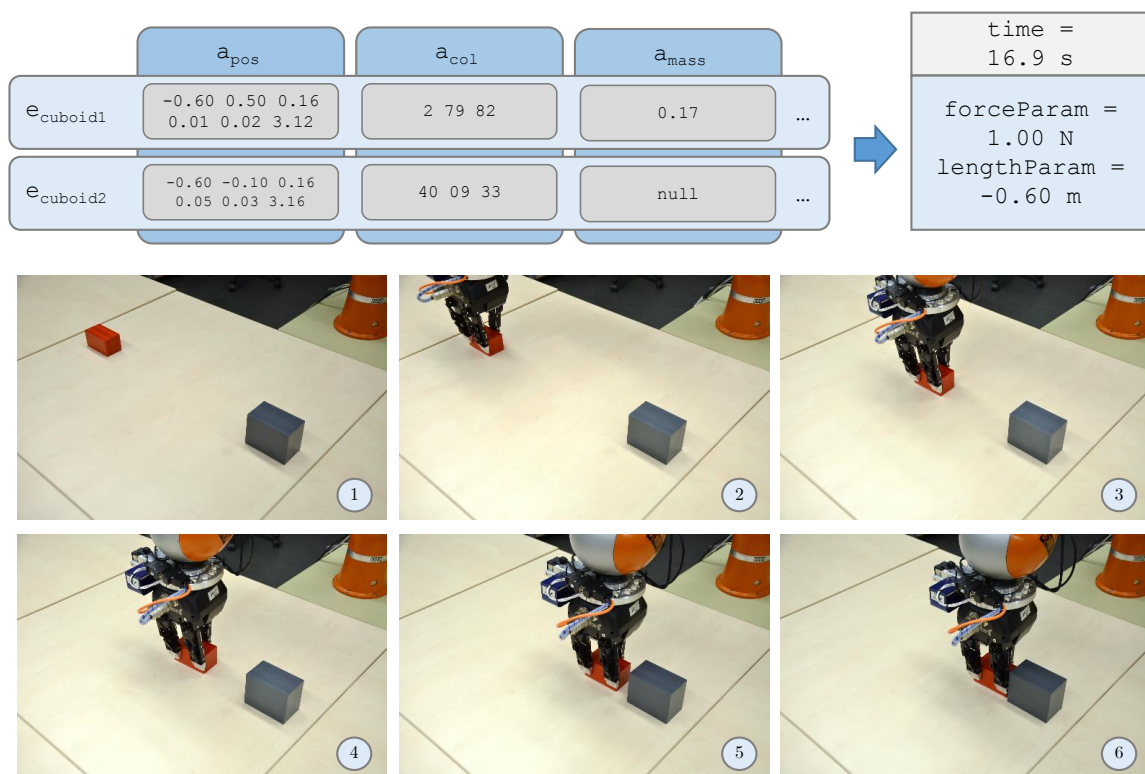


Abbildung 6.13: Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System2, Umgebung1 und Objektsatz2.

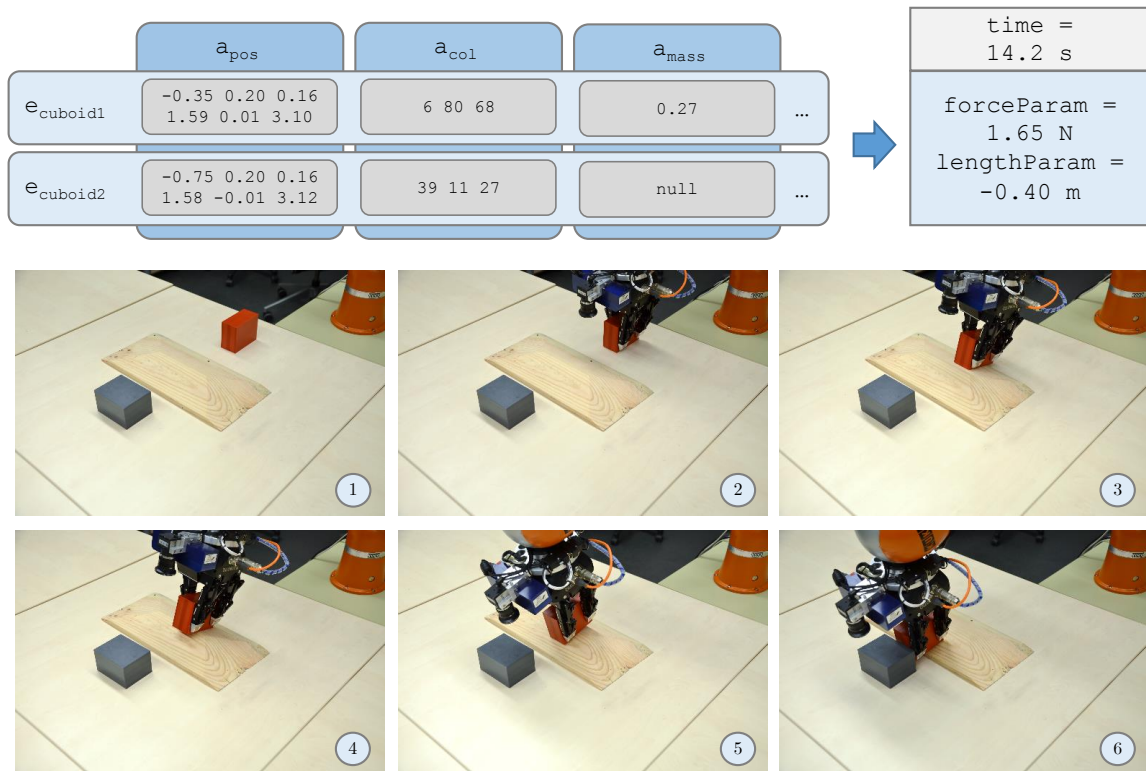


Abbildung 6.14: Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System2, Umgebung2 und Objektsatz1.

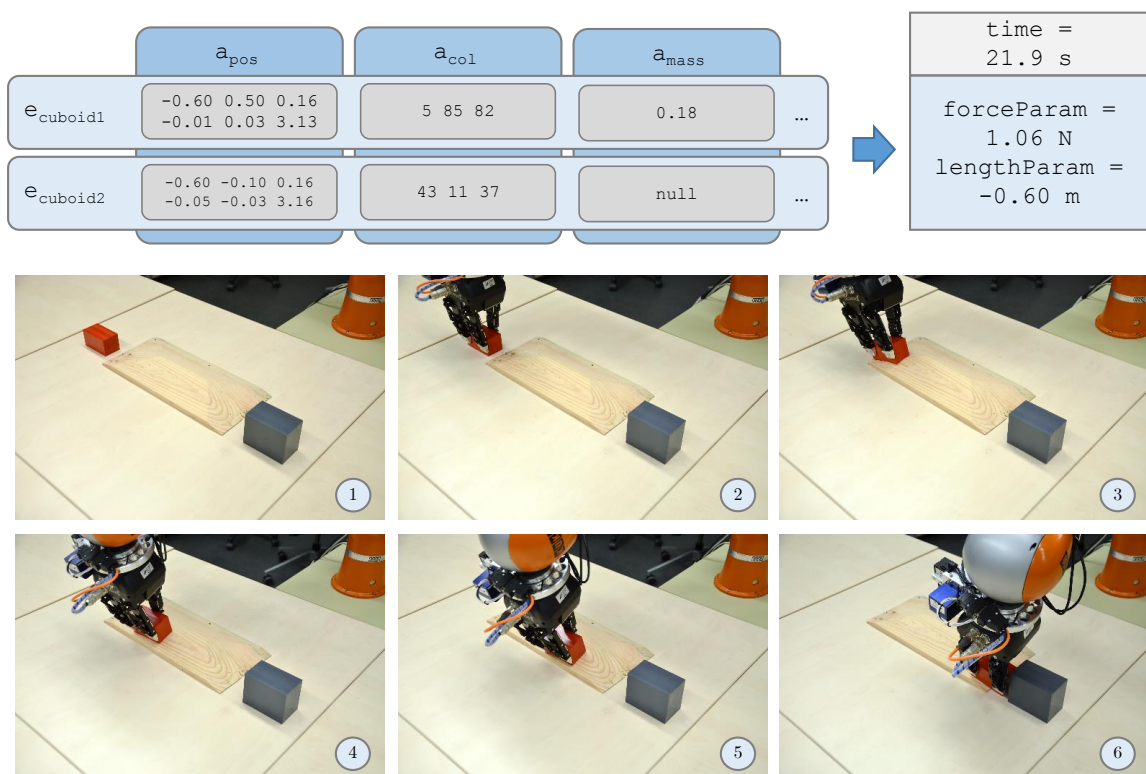


Abbildung 6.15: Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System2, Umgebung2 und Objektsatz2.

Analysiert man die Ausführungszeiten sowie die ermittelten Parameterwerte der einzelnen Varianten, so lassen sich eine Reihe individueller Merkmale und Abweichungen gegenüber den Stammdaten feststellen, welche im Folgenden diskutiert werden. Da die Parameterextraktion durch die im System hinterlegten Extraktionskomponenten durchgeführt wird, deren Umsetzung nicht im Fokus dieser Arbeit steht, werden die beobachteten Abweichungen ausschließlich aufgezeigt und mögliche Ursachen diskutiert. Daraus resultierende Optimierungen der entsprechenden Sensorik und den Extraktionskomponenten werden im Rahmen dieser Arbeit nicht durchgeführt (siehe Kapitel 1.3).

Zunächst werden die Ausführungszeiten der einzelnen Umgebungsvarianten analysiert und verglichen. Hier lässt sich erkennen, dass die Ausführung der instruierten Anweisung unter Nutzung der Umgebungsvariante **Umgebung1** weniger Laufzeit beansprucht als unter Nutzung der Umgebungsvariante **Umgebung2**. Dies kann unter anderem anhand der unterschiedlichen Wegstrecken erklärt werden, welche aufgrund der verschiedenen Umgebungsvarianten zurückgelegt werden. So entspricht beispielsweise unter Nutzung von **Objektsatz1** die Wegstrecke von **Umgebung1** etwa 90% der Wegstrecke von **Umgebung2** (siehe Abbildung 6.16). Einen weiteren Einfluss stellt die unterschiedliche Regelung des Kraftsollwerts dar. Während dieser für **Umgebung1** einmal erreicht und anschließend über den gesamten Ausführungszeitraum konstant gehalten werden kann, muss dieser unter Nutzung von **Umgebung2** kontinuierlich in Form einer Ausgleichsbewegung angepasst werden.

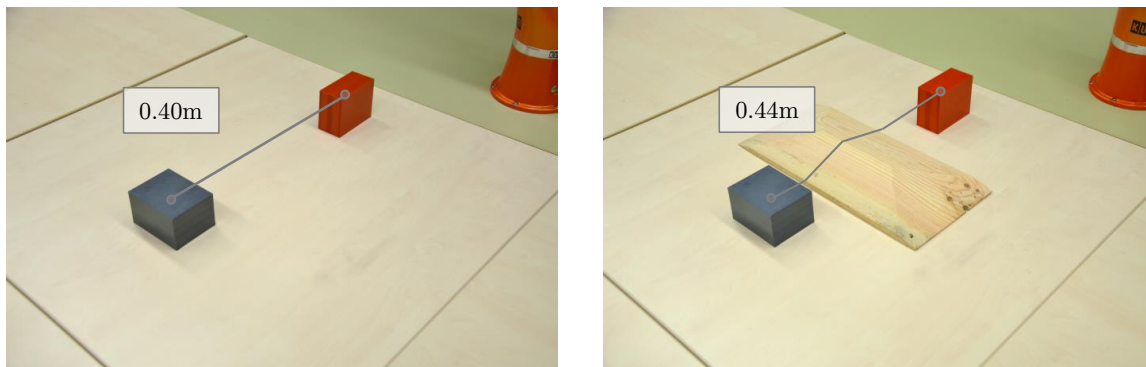


Abbildung 6.16: Visualisierung der unterschiedlichen Wegstrecken unter Nutzung der Umgebungsvarianten **Umgebung1** und **Umgebung2**, exemplarisch dargestellt für die weiteren Parameter **System1** und **Objektsatz1**.

Betrachtet man die durch die Extraktionskomponenten ermittelten und die daraus berechneten Aktionsparameter, so lassen sich die größten Abweichungen im Bereich der Farbbestimmung erkennen. Ein Grund hierfür ist beispielsweise die Abhängigkeit der vorhandenen Beleuchtungsverhältnisse. Betrachtet man exemplarisch die unterschiedlichen Verhältnisse wie in Abbildung 6.17 dargestellt, so liefert die verwendete Extraktionskomponente deutlich unterschiedliche Ergebnisse, insbesondere für das rote Testobjekt.

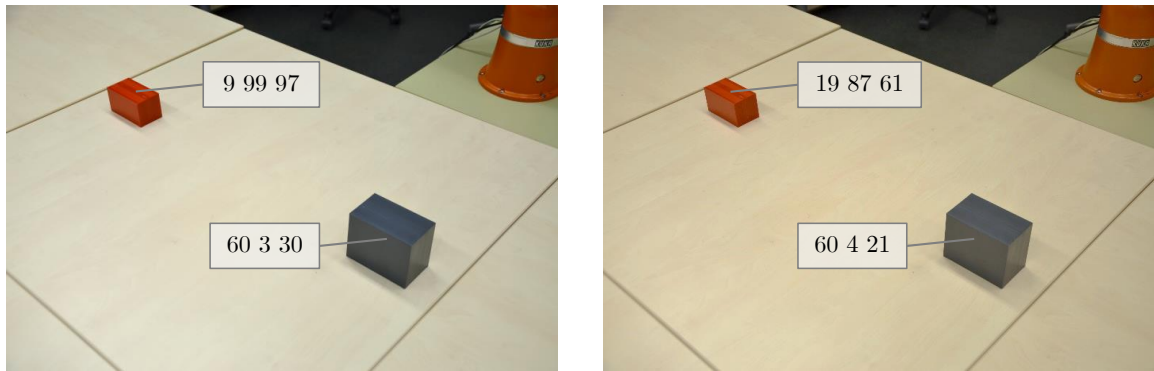


Abbildung 6.17: Visualisierung der Farbbestimmung für unterschiedliche Beleuchtungsverhältnisse unter Nutzung der Umgebungsvarianten *Umgebung1*, *System1* und *Objektsatz1*.

Für die extrahierten dynamischen und geometrischen Größen und die daraus resultierenden Aktionsparameter `forceParam` beziehungsweise `lengthParam` lassen sich im Allgemeinen geringe Abweichungen gegenüber den im Vorfeld bekannten Größen feststellen. Die vorhandenen Abweichungen werden beispielsweise einer fehlerbehafteten Kalibrierung der zur Extraktion verwendeten Sensorik zugeordnet.

Im Allgemeinen lässt sich festhalten, dass die im System verfügbare Sensorik und Extraktionskomponenten die Qualität der ermittelten Parametrisierung der auszuführenden Aktion beeinflussen. Diese werden daher unabhängig von der Aktionsbeschreibung in Form von Aktoren im System hinterlegt. Durch die einheitliche Schnittstelle können entsprechende Komponenten verwendet, beziehungsweise eingebunden werden, welche auf die Extraktion entsprechender Größen spezialisiert sind.

Abschließend lässt sich festhalten, dass die instruierte Anweisung für alle untersuchten Umweltvarianten erfolgreich ausgeführt werden kann. Die verwendeten Umweltvarianten betrachten die Variation des Robotersystems, welche sowohl individuelle Hardware- als auch Softwarekomponenten umfasst. Die weiteren Varianten umfassen die Variation der Roboterumgebung, welche in Form der sensorbasierten Aktionsausführung berücksichtigt wird sowie die Variation der Objekteigenschaften, welche in Form der extrahierten subsymbolischen Parameter sowie im Rahmen der Parametrisierung des auszuführenden Aktionsprimitivnetzes berücksichtigt werden. Als Basis der Ausführung dient die Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten, welche für alle untersuchten Umweltvarianten gemeinsam verwendet wird. Aufgrund der erfolgreichen Ausführung aller Umweltvarianten lässt sich festhalten, dass die im Rahmen dieser Arbeit vorgeschlagene Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten unabhängig von einer speziellen Roboterumgebung spezifiziert und flexibel in verschiedenen Systemen eingesetzt werden kann.

6.3 Benutzerevaluation

Wie in der Einleitung dieses Kapitels beschrieben, soll im Rahmen einer Benutzerevaluation die wissenschaftliche Fragestellung **F4** untersucht werden, welche wie folgt lautet:

- F4** In wie weit kann basierend auf den vorgeschlagenen Konzepten ein intuitiv zu verwendendes Gesamtsystem umgesetzt werden?

Dies umfasst zum einen die Bewertung der durch die domänenspezifische Sprache beschriebene symbolische Benutzerschnittstelle (Teil A) und zum anderen die durch einen Verbalisierten Physikalischen Effekt beschriebene Ausführung einer instruierten Anweisung (Teil B).

Im Rahmen der Benutzerevaluation werden Daten von insgesamt 32 Probanden untersucht und ausgewertet. Die Probanden werden hierzu gleichmäßig in vier Gruppen eingeteilt, wobei das Spektrum vom Standardbenutzer ohne Expertenwissen im Bereich der Robotik bis hin zum Robotikexperten reicht. Ein Proband wird entweder Teil A oder B der Benutzerevaluation zugewiesen. Die detaillierte Einteilung der Gruppen lautet wie folgt:

- *Gruppe A.*

Diese Gruppe umfasst die Personen mit *Allgemeinbildung*. Die Personen dieser Gruppe verfügen insbesondere kein Expertenwissen in den Bereichen Naturwissenschaften, Programmierung oder Robotik. Diese Gruppe stellt somit den Standardbenutzer dar.

- *Gruppe B.*

Diese Gruppe umfasst Personen mit Kenntnissen im Bereich *Naturwissenschaften*, beziehungsweise der Physik. Die Personen dieser Gruppe verfügen insbesondere kein Expertenwissen im Bereich der Programmierung oder der Robotik.

- *Gruppe C.*

Diese Gruppe umfasst Personen mit Kenntnissen im Bereich der *Informatik*, insbesondere in der Softwareentwicklung, beziehungsweise Programmierung. Diese Gruppe verfügt insbesondere kein Expertenwissen im Bereich der Robotik.

- *Gruppe D.*

Diese Gruppe umfasst Personen mit Kenntnissen im Bereich der *Robotik*, insbesondere in der Programmierung von Robotersystemen.

Im Teil A der Benutzerevaluation soll die intuitive Nutzung der symbolischen Benutzerschnittstelle basierend auf der domänenspezifischen Sprache evaluiert werden. Bei der Durchführung wird einem Probanden zunächst eine Ausführung einer Objektmanipulationsaufgabe vorgeführt. Im Anschluss soll ein Proband eine Anweisung unter Nutzung natürlichsprachlicher Symbole instruieren, welche die gezeigte Ausführung in Form eines Kommandos, beziehungsweise eines Imperativsatzes beschreibt. Dies wird für eine Reihe von Aufgaben aus dem Bereich der Objektmanipulation durchgeführt.

Als Erwartung wird formuliert, dass die Struktur der instruierten Anweisungen im Allgemeinen der Struktur entspricht, welche im Rahmen der domänenspezifischen Sprache verwendet wird. Das heißt zum einen, dass die verwendete verbale Komponente dem zu einem Verbalisierten Physikalischen Effekt zugeordneten Verb entspricht, welches bereits auf Basis empirischer Daten ermittelt wurde (siehe Kapitel 3.3). Zum anderen heißt dies, dass als Parameterphrasen die entsprechenden obligatorischen Ergänzungen in geeigneten Ausdrucksformen verwendet werden.

Im Teil B der Benutzerevaluation soll die intuitive Ausführung einer symbolischen Anweisung untersucht werden, welche auf Basis der domänenspezifischen Sprache formuliert und unter Nutzung der Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten ausgeführt wird. Zur Durchführung werden den Probanden eine Reihe von Anweisungen vorgelegt, welche gemäß der domänenspezifischen Sprache formuliert werden. Im Anschluss sollen die Probanden die erwartete Ausführung des Robotersystems in Form einer Demonstration an einem Robotersystem beschreiben. Hierzu wird ein Robotersystem verwendet, welches durch Führung programmiert werden kann und entsprechende subsymbolische Parameter wie beispielsweise die aktuelle Roboterposition während der Demonstration aufzeichnet. Die erhobenen Daten werden abschließend mit der Ausführung unter Nutzung der Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten verglichen und qualitativ bewertet.

Als Erwartung wird formuliert, dass sich die erfassten Demonstrationen einer Aufgabe im Allgemeinen nicht wesentlich zur entsprechenden Ausführung basierend auf Verbalisierten Physikalischen Effekten unterscheiden. Dies sollte insbesondere für die Manipulationsphase zutreffen, in welcher die Manipulation der Objekteigenschaften stattfindet und welche daher einen wesentlichen Beitrag der semantischen Information darstellt.

Im Rahmen der Benutzerevaluation werden exemplarisch die Aktionen verwendet, welche in Tabelle 6.1 dargestellt sind. Die Auswahl beinhaltet mindestens einen Vertreter für jede Art der im Rahmen dieser Arbeit untersuchten elementaren physikalischen Effekte (siehe Kapitel 3.2).

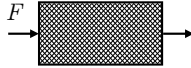
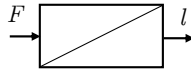
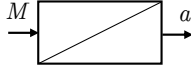
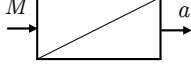
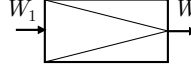
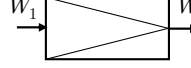
Aktion	Verwendung in der domänenspezifischen Sprache	Verbalisierter Physikalischer Effekt
berühre / touch	<p>Allgemeine Verwendung:</p> <p>berühre Nominalphrase</p> <p>touch Nominalphrase</p> <p>Beispielanweisung:</p> <p>berühre den roten Klotz</p> <p>touch the red block</p>	 <p>(touch, absorb, (F, N, force), \emptyset, MPN_{touch})</p>
schiebe / shove	<p>Allgemeine Verwendung:</p> <p>schiebe Nominalphrase Präpositionalphrase</p> <p>shove Nominalphrase Präpositionalphrase</p> <p>Beispielanweisung:</p> <p>schiebe das rote Objekt 30 cm nach rechts</p> <p>shove the red object for 30 cm to the right</p>	 <p>(shove, transform, (F, N, force), (l, m, length), MPN_{shove})</p>
drehe / rotate	<p>Allgemeine Verwendung:</p> <p>drehe Nominalphrase Präpositionalphrase</p> <p>rotate Nominalphrase Präpositionalphrase</p> <p>Beispielanweisung:</p> <p>drehe den roten Quader um 90 Grad</p> <p>rotate the red cuboid for 90 degrees</p>	 <p>(rotate, transform, (M, Nm, torque), (a, rad, angle), MPN_{rotate})</p>
kippe / tilt	<p>Allgemeine Verwendung:</p> <p>kippe Nominalphrase Präpositionalphrase</p> <p>tilt Nominalphrase Präpositionalphrase</p> <p>Beispielanweisung:</p> <p>kippe den Klotz um 45 Grad</p> <p>tilt the block for 45 degrees</p>	 <p>(tilt, transform, (M, Nm, torque), (a, rad, angle), MPN_{tilt})</p>
hebe / lift	<p>Allgemeine Verwendung:</p> <p>hebe Nominalphrase Präpositionalphrase</p> <p>lift Nominalphrase Präpositionalphrase</p> <p>Beispielanweisung:</p> <p>hebe die rote Box um 30 cm</p> <p>lift the red box for 30 cm</p>	 <p>(lift, change, (W₁, J, energy), (W₂, J, energy), MPN_{lift})</p>
senke / lower	<p>Allgemeine Verwendung:</p> <p>senke Nominalphrase Präpositionalphrase</p> <p>lower Nominalphrase Präpositionalphrase</p> <p>Beispielanweisung:</p> <p>senke die rote Box um 0.3 m</p> <p>lower the red box for 0.3 m</p>	 <p>(lower, change, (W₁, J, energy), (W₂, J, energy), MPN_{lower})</p>

Tabelle 6.1: Darstellung der im Rahmen der Benutzerevaluation verwendeten Aktionen inklusive der Verwendung gemäß der domänenspezifischen Sprache sowie dem zugeordneten Verbalisierten Physikalischen Effekt.

Teil A - Evaluation der symbolischen Benutzerschnittstelle

Im Folgenden werden zunächst die im Rahmen von Teil A der Benutzerevaluation gewonnenen Erkenntnisse beschrieben und diskutiert. Basierend auf den Daten von 16 Probanden wurden die Ergebnisse ermittelt, welche in Tabelle 6.2 dargestellt sind.



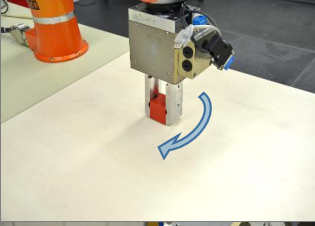

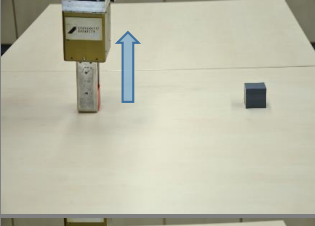
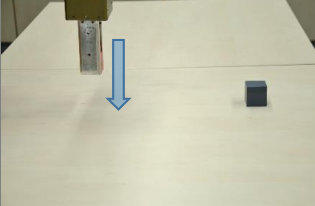
Aktion	Ausführung / Demonstration	Anweisungs- struktur	Ergebnisse				
			Gruppe A-D				Total
			r _A	r _B	r _C	r _D	r _T
berühre / touch		berühre NP	1.00	0.75	1.00	0.50	0.82
		berühre NP PP	0.00	0.00	0.00	0.50	0.12
		drücke NP	0.00	0.25	0.00	0.00	0.06
schiebe / shove		schiebe NP PP	1.00	0.75	0.75	1.00	0.88
		drücke NP PP	0.00	0.25	0.00	0.00	0.06
		führe NP PP	0.00	0.00	0.25	0.00	0.06
drehe / rotate		drehe NP PP	1.00	1.00	1.00	0.75	0.94
		rotiere NP PP	0.00	0.00	0.00	0.25	0.06
kippe / tilt		kippe NP PP	0.75	1.00	1.00	1.00	0.94
		neige NP PP	0.25	0.00	0.00	0.00	0.06
hebe / lift		hebe NP PP	0.75	0.75	1.00	1.00	0.88
		bewege NP PP	0.25	0.25	0.00	0.00	0.12
senke / lower		senke NP PP	0.75	0.75	0.75	1.00	0.82
		bewege NP PP	0.25	0.25	0.25	0.00	0.18

Tabelle 6.2: Darstellung der im Rahmen von Teil A der Benutzerevaluation ermittelten Daten basierend auf einer jeweiligen Demonstration der auszuführenden Aktion.

In der tabellarischen Übersicht sind die auf Basis einer Demonstration erfassten Anweisungsstrukturen für die in Tabelle 6.1 beschriebenen auszuführenden Aktionen dargestellt. Für jede Aktion und Probandengruppe wird eine Ratio r_i mit $i \in \{A, B, C, D, T\}$ bestimmt, welche folgendes Verhältnis beschreibt:

$$r_i = \frac{\text{Anzahl der Nennungen einer Anweisungsstruktur}}{\text{Anzahl der Probanden einer Gruppe } |i|}$$

Im Idealfall kann somit $r_T = 1.00$ für eine auszuführende Aktion festgestellt werden. In diesem Fall verwendet jeder Proband eine spezifische Anweisungsstruktur zur Instruktion der auszuführenden Aufgabe. Im Folgenden werden die Ergebnisse der einzelnen Aufgaben untersucht und insbesondere die Abweichungen gegenüber der im Rahmen der domänenspezifischen Sprache beschriebenen Struktur analysiert und diskutiert.

Im ersten Testfall wurde die Ausführung einer Anweisung demonstriert, welche gemäß der Form **berühre NP** im Rahmen der domänenspezifischen Sprache formuliert werden kann. Hierfür lässt sich eine Anweisungsstruktur erkennen, welche von 13 der 16 Probanden ($r_T = 0.82$) zur Beschreibung verwendet wird. Diese entspricht der Form **berühre NP** und somit der im Rahmen der domänenspezifischen Sprache verwendeten Anweisungsstruktur. Insbesondere lässt sich festhalten, dass die Gruppe der Personen ohne Expertenwissen (Gruppe A) ausschließlich die im Rahmen der domänenspezifischen Sprache verwendete Anweisungsstruktur zur Instruktion nutzt.

Es lassen sich zudem zwei weitere Varianten der Anweisungsstruktur feststellen. Diese sind zum einen **berühre NP PP** und zum anderen **drücke NP**. Bei der Variante **berühre NP PP** wird zur Instruktion eine zusätzliche, fakultative Ergänzung in Form einer Präpositionalphrase instruiert. Eine der verwendeten Anweisungen lautet beispielsweise *berühre den roten Würfel an der Oberseite*. Hier wird somit beispielsweise die Lage der Berührung genauer spezifiziert. Im Rahmen der ausgewerteten Daten ist diese zusätzliche Angabe von Information ausschließlich in den formulierten Anweisungen der Robotikexperten (Gruppe D) zu beobachten. Diese Gruppe besitzt Kenntnisse über die zur Ausführung einer Roboteraktion benötigten Parameter, welche beispielsweise die exakte Spezifikation der Zielposition umfassen. Daher wird vermutet, dass ein solcher Benutzer dazu neigt, zusätzliche Information im Rahmen der instruierten Anweisung explizit, beziehungsweise genauer zu spezifizieren. Daraus wird die Schlussfolgerung abgeleitet, dass die im Rahmen der domänenspezifischen Sprache beschriebene Anweisungsstruktur um optionale, fakultative Ergänzungen erweitert werden kann, um die Flexibilität der Anweisungsinstruktion zu erhöhen. Dies stellt somit eine potentielle Erweiterung dar, welche im Rahmen einer Weiterentwicklung des Gesamtsystems untersucht werden kann (siehe Kapitel 7.2).

Die weitere zur Instruktion verwendete Struktur lautet **drücke NP**. Hier wird zur Aktionsbeschreibung eine verbale Komponente verwendet, welche als Synonym zum erwarteten Ausdruck **berühre NP** genutzt werden kann. Die verbale Komponente kann jedoch nicht eindeutig einem auszuführenden Prinzipiellen Physikalischen Effekt zugeordnet werden. Dies wurde bereits im Rahmen der Symbolzuordnung der Verbalisierten Physikalischen Effekte beobachtet und beschrieben (siehe Kapitel 3.3).

Im zweiten Testfall wurde die Ausführung einer Anweisung demonstriert, welche gemäß der Form **schiebe NP PP** im Rahmen der domänenspezifischen Sprache formuliert werden kann. Für diese Demonstration lässt sich eine Anweisungsstruktur erkennen, welche von 14 der 16 Probanden ($r_T = 0.88$) zur Beschreibung verwendet wird. Diese entspricht der Form **schiebe NP PP** und somit der im Rahmen der domänenspezifischen Sprache verwendeten Anweisungsstruktur. Insbesondere lässt sich festhalten, dass auch hier die Gruppe der Personen ohne Expertenwissen (Gruppe A) ausschließlich die im Rahmen der domänenspezifischen Sprache verwendete Anweisungsstruktur zur Instruktion nutzt. Auch hier lassen sich zwei weitere Varianten der Anweisungsstruktur feststellen. Diese sind zum einen **drücke NP PP** und zum anderen **führe NP**. Wie bei der Ausführung des ersten Testfalls bereits beschrieben, ist die verbale Beschreibung **drücke** nicht eindeutig einem Prinzipiellen Physikalischen Effekt zuzuordnen, da diese auch im Sprachgebrauch unterschiedlich verwendet wird (siehe Kapitel 3.3).

Im dritten Testfall wurde die Ausführung einer Anweisung demonstriert, welche gemäß der Form **drehe NP PP** im Rahmen der domänenspezifischen Sprache formuliert werden kann. Für diese Demonstration lässt sich eine Anweisungsstruktur erkennen, welche von 15 der 16 Probanden ($r_T = 0.94$) zur Beschreibung verwendet wird. Diese entspricht der Form **drehe NP PP** und somit der im Rahmen der domänenspezifischen Sprache verwendeten Anweisungsstruktur. Hier ist lediglich eine abweichende Struktur zu erkennen, welche jedoch mit **rotiere NP PP** eine als Synonym zu verwendende verbale Aktionsbeschreibung darstellt.

Im vierten Testfall wurde die Ausführung einer Anweisung demonstriert, welche gemäß der Form **kippe NP PP** im Rahmen der domänenspezifischen Sprache formuliert werden kann. Für diese Demonstration lässt sich eine Anweisungsstruktur erkennen, welche von 15 der 16 Probanden ($r_T = 0.94$) zur Beschreibung verwendet wird. Diese entspricht der Form **kippe NP PP** und somit der im Rahmen der domänenspezifischen Sprache verwendeten Anweisungsstruktur. Hier ist ebenfalls mit **neige NP PP** lediglich eine abweichende Struktur zu erkennen, welche von einem Probanden aus der Benutzergruppe ohne Expertenwissen formuliert wurde. Dies entspricht ebenfalls einer als Synonym zu verwendenden verbalen Beschreibung der auszuführenden Aktion.

Im fünften Testfall wurde die Ausführung einer Anweisung demonstriert, welche gemäß der Form **hebe NP PP** im Rahmen der domänenspezifischen Sprache formuliert werden kann. Für diese Demonstration lässt sich eine Anweisungsstruktur erkennen, welche von 14 der 16 Probanden ($r_T = 0.88$) zur Beschreibung verwendet wird. Diese entspricht der Form **hebe NP PP** und somit der im Rahmen der domänenspezifischen Sprache verwendeten Anweisungsstruktur. Es lässt sich zudem eine alternative Anweisungsstruktur feststellen, welche durch **bewege NP PP** gegeben ist.

Die verbale Umschreibung **bewege** stellt hierbei eine allgemeine Form, **hebe** einen Spezialfall der Lageänderung dar. Wird als Anweisungsstruktur die Form **bewege NP PP** verwendet, so lautet eine der konkreten Anweisungen beispielsweise *bewege den roten Würfel 30 cm nach oben*. Hier wird somit explizit die Richtung spezifiziert, welche in der Verwendung der verbalen Umschreibung **hebe** implizit in der semantischen Beschreibung vorhanden ist.

Im letzten Testfall wurde die Ausführung einer Anweisung demonstriert, welche gemäß der Form **senke** *NP PP* im Rahmen der domänenspezifischen Sprache formuliert werden kann. Für diese Demonstration lässt sich eine Anweisungsstruktur erkennen, welche von 13 der 16 Probanden ($r_T = 0.82$) zur Beschreibung verwendet wird. Diese entspricht der Form **senke** *NP PP* und somit der im Rahmen der domänenspezifischen Sprache verwendeten Anweisungsstruktur. Auch hier lässt die alternative Anweisungsstruktur **bewege** *NP PP* beobachten, welche die allgemeine Form der Lageänderung beschreibt. Eine zugehörige konkrete Anweisung lautet beispielsweise *bewege den roten Würfel 30 cm nach unten*.

Abschließend lässt sich festhalten, dass für alle untersuchten Testfälle eine dominante Anweisungsstruktur beobachtet werden konnte, welche von mindestens 13 der 16 Probanden zur Beschreibung einer demonstrierten Anweisungsausführung genutzt wurde. Diese entspricht für alle untersuchten Aktionen der Struktur, welche im Rahmen der domänenspezifischen Sprache beschrieben ist. Somit konnte die im Vorfeld formulierte Erwartung anhand der empirischen Daten verifiziert werden. Daher wird abschließend gefolgert, dass die im Rahmen der domänenspezifischen Sprache verwendete Abstraktion der Form

Aktion *Parameterphrase₁* ... *Parameterphrase_n*

mit den zu einem Verbalisierten Physikalischen Effekt zugeordneten verbalen Ausdrücken als Schlüsselworte und den dazugehörigen obligatorischen Ergänzungen in geeigneten Ausdrucksformen als Parameterphrasen eine symbolische und intuitive Benutzerschnittstelle umgesetzt werden kann.

Teil B - Evaluation der Aktionsausführung

Wie zu Beginn dieses Kapitels dargestellt, wird in diesem Teil der Benutzerevaluation die intuitive Aktionsausführung basierend auf Verbalisierten Physikalischen Effekten untersucht. Hierzu werden den Probanden zunächst eine Reihe an Anweisungen vorgelegt, welche gemäß der domänenspezifischen Sprache formuliert werden, das heißt eine auszuführende Aktion wird in Form eines verbalen Ausdrucks und den entsprechenden obligatorischen Ergänzungen in geeigneten Ausdrucksformen spezifiziert. Als Testfälle werden weiterhin die Aktionen verwendet, welche in Tabelle 6.1 dargestellt sind und bereits im Teil A der Benutzerevaluation verwendet wurden.

Um die Vergleichbarkeit zwischen den einzelnen Demonstrationen der Probanden zu gewährleisten, wird eine einheitliche Roboterumgebung verwendet (siehe Abbildung 6.18, links). Zudem werden allen Probanden die identischen auszuführenden Anweisungen genannt, welche in Abbildung 6.18 (rechts) dargestellt sind.

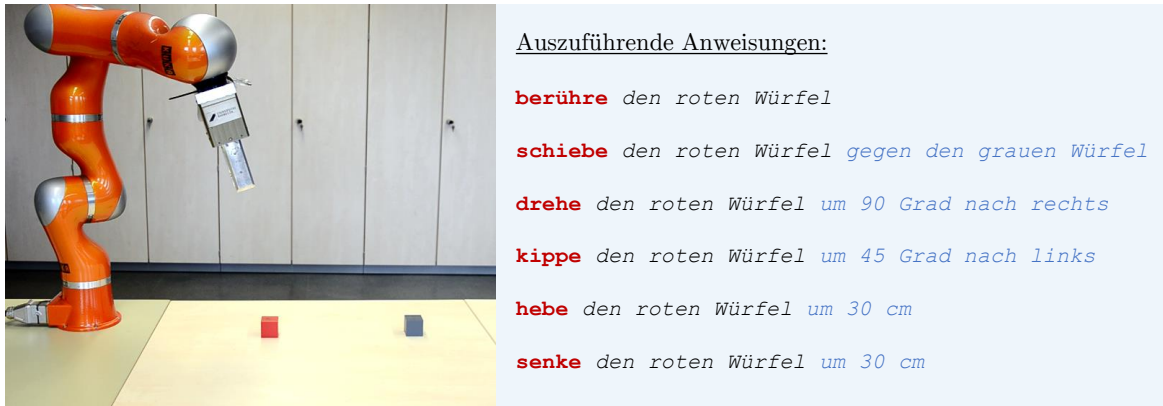


Abbildung 6.18: Im Rahmen von Teil B der Benutzerevaluation verwendete Umgebung (links) und auszuführende Anweisungen gemäß der domänenspezifischen Sprache (rechts).

Im Folgenden werden die im Rahmen von Teil B der Benutzerevaluation gewonnenen Erkenntnisse beschrieben und diskutiert. In den Abbildungen 6.19 - 6.24 sind die Ergebnisse für die einzelnen auszuführenden Anweisungen dargestellt.

Bevor die Ergebnisse der einzelnen Aktionen diskutiert werden, wird die grundsätzliche Struktur der Abbildungen 6.19 - 6.24 beschrieben. In jeder Abbildung ist auf der linken Seite eine exemplarische Demonstration der auszuführenden Anweisung durch eine der Testpersonen visualisiert. Auf der rechten Seite einer entsprechenden Abbildung sind die im Rahmen der Demonstrationen instruierten Roboterbahnen dargestellt. Zum einen ist hier jeweils die resultierende, aus allen Demonstrationen der Probanden gemittelte Bahn dargestellt (grau). Zum anderen ist die Bahn abgebildet, welche durch die Ausführung basierend auf Verbalisierten Physikalischen Effekten ermittelt wird (blau). Zur Visualisierung der Bahnen wird im Allgemeinen die x- und z-Komponente der Endeffektorstellung verwendet. Die y-Komponente der Endeffektorstellung kann aufgrund der gewählten Roboterumgebung, beziehungsweise der entsprechenden Startkonfiguration sowie der Objektpositionen vernachlässigt werden.

Zur Zuordnung der einzelnen Phasen der Objektmanipulationsaufgabe sind diese entsprechend farblich hinterlegt. Zudem sind die in der linken Seite abgebildeten Schritte der Demonstration den Ausführungsphasen, beziehungsweise dem Ausführungsfortschritt zugeordnet.

Die ermittelten Ergebnisse für die Ausführung der Instruktion **berühre den roten Würfel** sind in Abbildung 6.19 dargestellt. Wie im rechten Teil der Abbildung zu erkennen, unterscheidet sich die Annäherungsbewegung zwischen den Demonstrationen und der Ausführung basierend auf Verbalisierten Physikalischen Effekten. Während die Ausführung basierend auf Verbalisierten Physikalischen Effekten den direkten Weg hin zum entsprechenden Objekt ausführt und versucht den Kontakt auf direkten Weg in Richtung des Objektmittelpunktes herzustellen, neigen die Testpersonen zur Ausführung einer Transferbewegung um das Objekt an der Objektoberseite zu berühren, was beispielsweise auch der exemplarischen Demonstration auf der linken Seite der Abbildung zu beobachten ist.

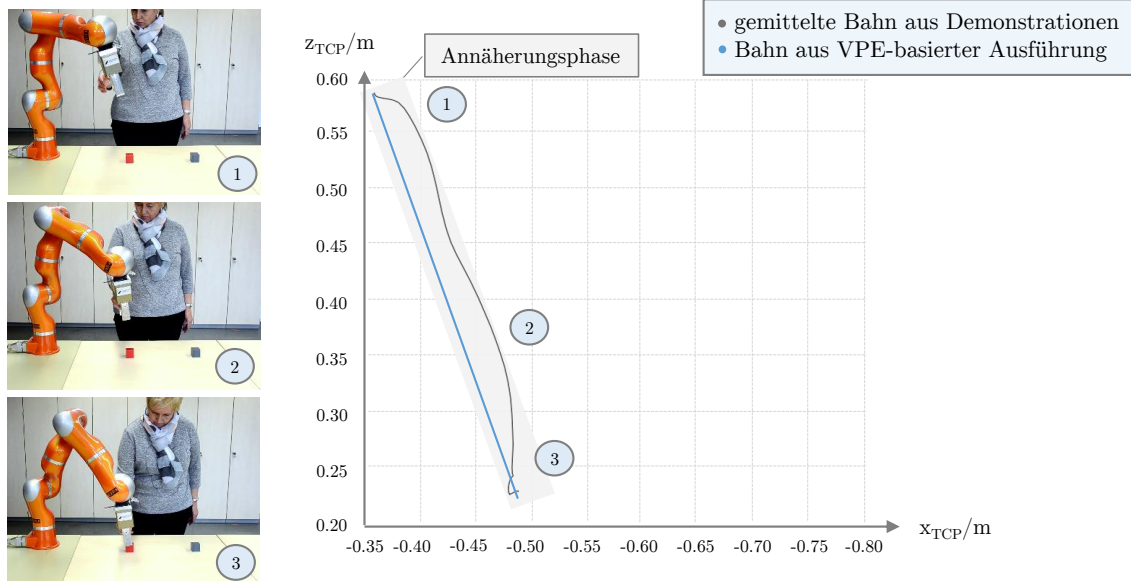


Abbildung 6.19: Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung **berühre den roten Würfel**.

Die ermittelten Ergebnisse für die Ausführung der Instruktion **schiebe den roten Würfel gegen den grauen Würfel** sind in Abbildung 6.20 dargestellt. Im Gegensatz zur zuvor betrachteten Aktion ist hier eine entsprechende Manipulationsphase vorhanden, das heißt das Objekteigenschaften während der Aktionsausführung verändert werden, beispielsweise die Objektposition. Aus der resultierenden Bahn der Annäherungsphase lässt sich erkennen, dass die im Rahmen der vorangegangenen Aktion beobachtete Transferbewegung nicht so stark ausgeprägt ist. Betrachtet man die demonstrierten Bahnen der Manipulationsphase, so lässt sich hier eine geringe Abweichung gegenüber der Aktionsausführung basierend auf Verbalisierten Physikalischen Effekten feststellen. Während diese den Kontakt zwischen Objekt und Umgebung kontinuierlich aufrecht erhält, kann bei der gemittelten Demonstration der Probanden beobachtet werden, dass der Kontakt nicht kontinuierlich eingehalten wird, was beispielsweise auch in Bild 2 der im linken Bereich der Abbildung dargestellten Demonstration zu erkennen ist.

Die ermittelten Ergebnisse für die Ausführung der Instruktion **drehe den roten Würfel um 90 Grad nach rechts** sind in Abbildung 6.21 dargestellt. Im Gegensatz zu den bisher betrachteten Aktionen, stellt eine wesentliche Komponente der Manipulationsphase dieser Aktion eine rotatorische Bewegung dar. Hierbei konnte allgemein beobachtet werden, dass die Demonstration von rotatorischen Bewegungen den Probanden deutlich schwerer fällt als die Demonstration von translatorischen Bewegungen. Zum einen hat dies zur Folge, dass nicht alle Demonstrationen zur Auswertung herangezogen werden können. Zum anderen ist dies jedoch eine Erkenntnis, welche in Systemen aus dem Bereich des Programmierens durch Vormachen weiter untersucht werden kann. Zusätzlich zur translatorischen Bewegung ist in der Abbildung die für die Aktion relevante rotatorische Bewegung aufgezeigt. Betrachtet man die ermittelten Bahnen für die Manipulationsphase, so ist hier eine ähnliche Zielposition festzustellen. Für die rotatorische Bewegung lassen sich jedoch ein unterschiedliche Ausführungsverläufe feststellen.

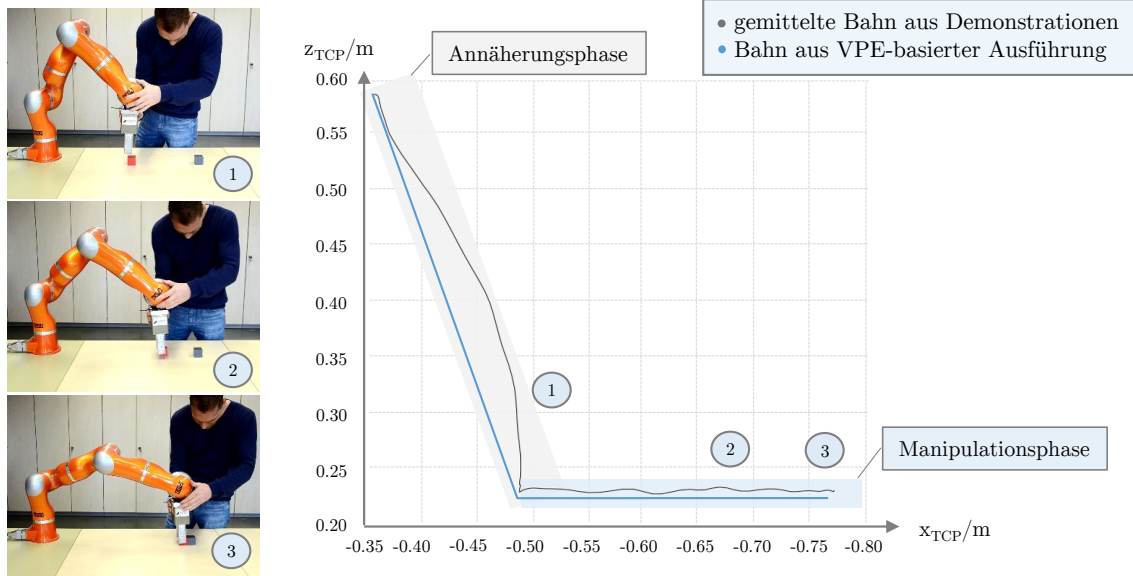


Abbildung 6.20: Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung **schiebe** den roten Würfel gegen den grauen Würfel.

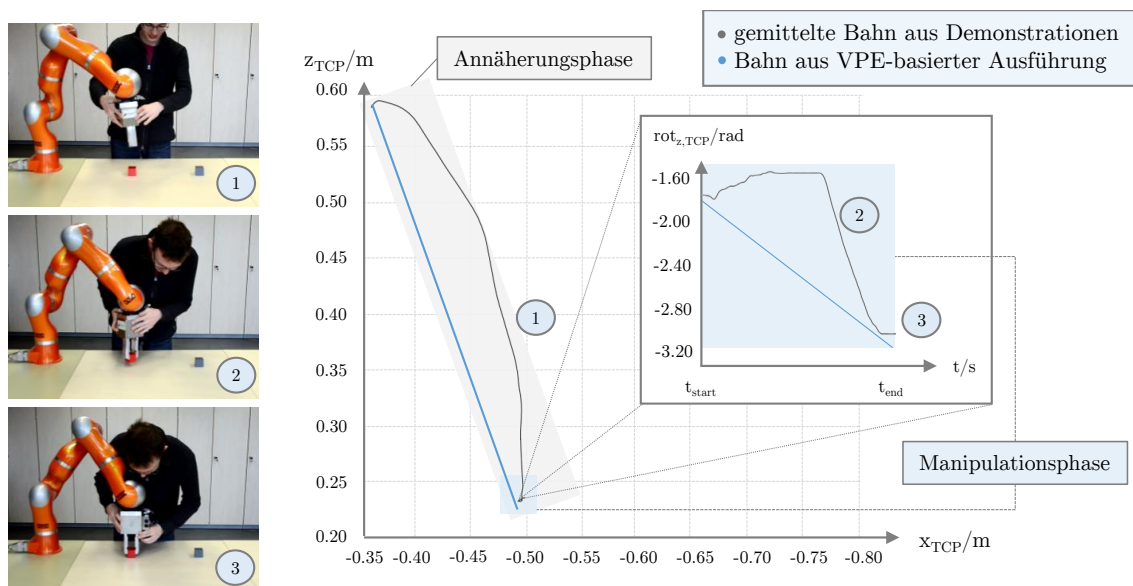


Abbildung 6.21: Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung **drehe** den roten Würfel um 90 Grad nach rechts. Zusätzlich dargestellt ist die während der Manipulationsphase relevante rotatorische Bewegung. Die dargestellten Zeiten t_{start} und t_{end} entsprechen den normierten Start- und Endzeitpunkten der Manipulationsphase.

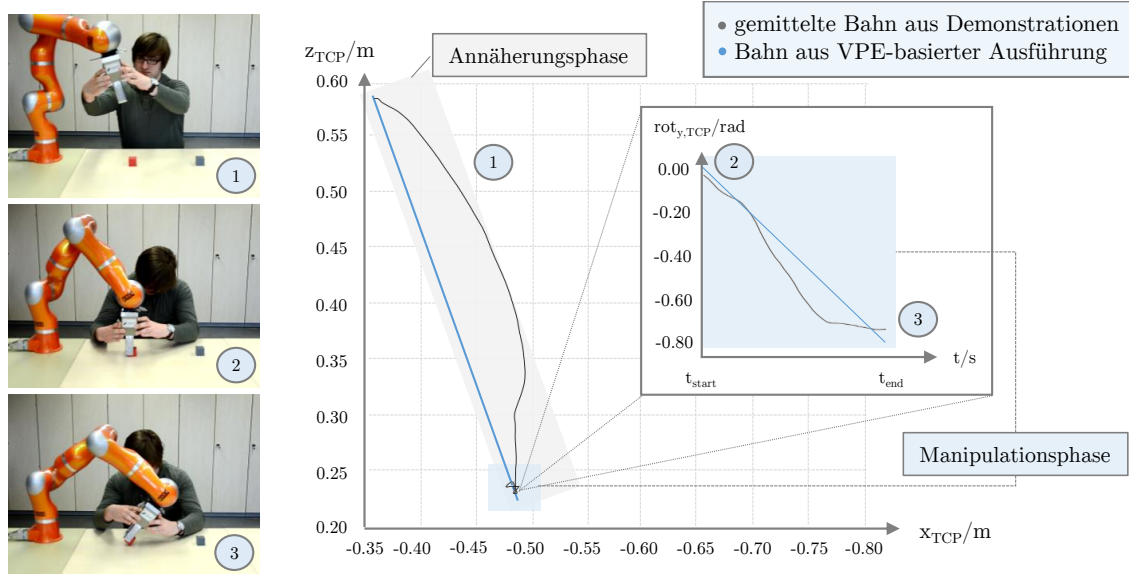


Abbildung 6.22: Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung **kippe den roten Würfel um 45 Grad nach links**. Zusätzlich dargestellt ist die während der Manipulationsphase relevante rotatorische Bewegung. Die dargestellten Zeiten t_{start} und t_{end} entsprechen den normierten Start- und Endzeitpunkten der Manipulationsphase.

Die ermittelten Ergebnisse für die Ausführung der Instruktion **kippe den roten Würfel um 45 Grad nach links** sind in Abbildung 6.22 dargestellt. Wie bereits in Tabelle 6.1 dargestellt, ist die Aktion *kippen* verwandt mit der im vorangegangenen Absatz beschriebenen Aktion *drehen*. Daher lassen sich auch hierfür identische Schwierigkeiten bei der Demonstration von rotatorischen Bewegungen feststellen. Betrachtet man die auf Basis der auswertbaren Demonstrationen ermittelte Bahn für die Manipulationsphase, so wurde der Unterschied zwischen den beiden Aktionen *drehen* und *kippen* intuitiv durch die Probanden erkannt. Für diese Aktion ist somit eine unterschiedliche rotatorische Bewegung relevant, welche im Rahmen der Abbildung dargestellt ist. Insgesamt können vergleichbare Ergebnisse zwischen den Demonstrationen und der Ausführung basierend auf Verbalisierten Physikalischen Effekten erkannt werden.

Die ermittelten Ergebnisse für die Ausführung der Instruktionen **hebe den roten Würfel um 30 cm** und **senke den roten Würfel um 30 cm** sind in Abbildung 6.23, beziehungsweise Abbildung 6.24 dargestellt. Diese werden im Rahmen der Benutzerevaluation sequenziell ausgeführt, das heißt die Endposition der Anweisungsausführung der Aktion *hebe* dient als Startkonfiguration für die Ausführung der Aktion *senke*. Da bei dieser Anweisung ausschließlich translatorische Bewegungen relevant sind, können hierfür alle Demonstrationen zur Auswertung genutzt werden. Betrachtet man die resultierende Bahn für die Annäherungsphase, so lässt sich für die Demonstrationen eine entsprechende Transferbewegung und daher eine Abweichung zur direkten Bewegungen bei der Aktionsausführung basierend auf Verbalisierten Physikalischen Effekten feststellen. Betrachtet man die Bahnen der Manipulationsphasen der beiden Aktionen, so sind die aufgrund der Demonstrationen ermittelten Bahnen ähnlich zu den auf Basis der Aktionsausführung durch Verbalisierte Physikalische Effekte.

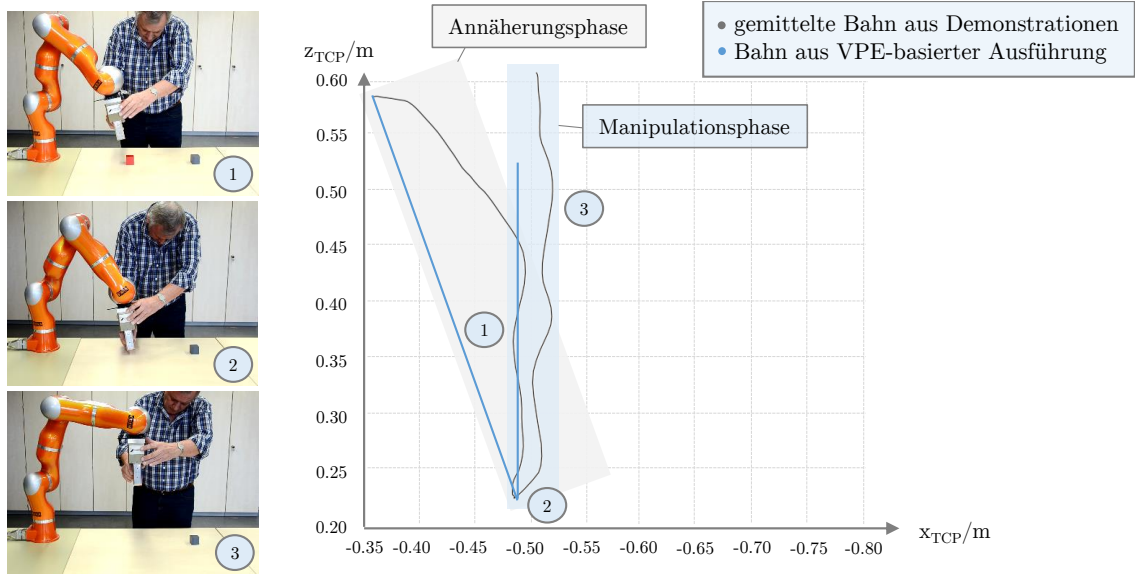


Abbildung 6.23: Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung **hebe** den roten Würfel um 30 cm.

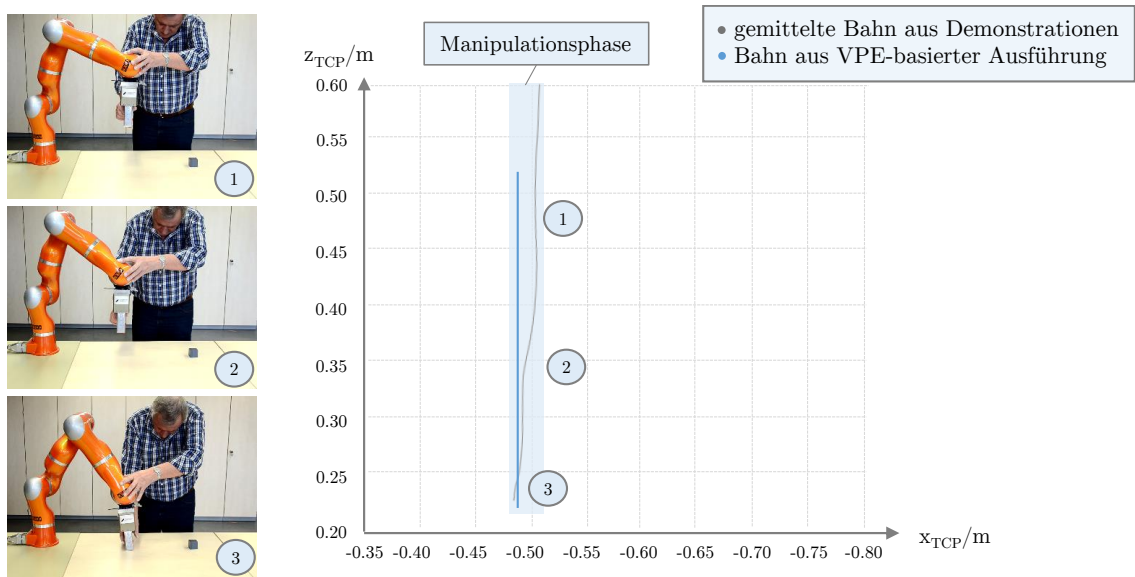


Abbildung 6.24: Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung **senke** den roten Würfel um 30 cm.

Abschließend lässt sich festhalten, dass alle Probanden die vorgegebenen Anweisungen verstanden und eine intuitive Vorstellung, beziehungsweise eine entsprechende Erwartungshaltung an die Ausführung der Aktion besitzen. Diese konnte für die Aktionen **berühre**, **schiebe**, **hebe** und **senke** von allen Probanden unter Nutzung eines Robotersystems demonstriert werden. Es wurde jedoch beobachtet, dass die Demonstration von rotatorischen Bewegungen, welche beispielsweise für die Aktionen **drehe** und **kippe** benötigt werden, teilweise nicht, beziehungsweise nur unter Hilfestellung demonstriert werden konnten.

Betrachtet man die aufgrund der Demonstrationen ermittelten Bahnen für die entsprechenden Ausführungsphasen einer Objektmanipulationsaufgabe und vergleicht diese mit den resultierenden Bahnen der Aktionsausführung basierend auf Verbalisierten Physikalischen Effekten, so sind diese im Allgemeinen vergleichbar. Die größten Abweichungen lassen sich im Allgemeinen während der Annäherungsphase feststellen. Hier neigen die Probanden zu einer Transferbewegung, während die Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten den direkten Weg hin zum manipulierenden Objekt ausführt. Betrachtet man die Bahnen der Manipulationsphase, so sind die grundsätzlichen Bewegungen nahezu identisch, die resultierenden Abweichungen sind typischerweise aufgrund einer ungenauen Platzierung oder der unpräzisen Führung des Robotersystems vorhanden. Insgesamt liegen die festgestellten Abweichungen im Bereich von wenigen Zentimetern.

Zum Eingang dieses Kapitels wurde die Erwartung formuliert, dass die durch die Ausführung von Verbalisierten Physikalischen Effekte beschriebenen Roboteraktionen der intuitiven Ausführung entspricht, welche eine Person vom Robotersystem erwartet. Insbesondere sollte dies für die Ausführung der Manipulationsphase gültig sein, da diese die Änderung der Objekteigenschaften und daher einen wesentlichen Teil der Semantik einer auszuführenden Aktion dargestellt. Anhand der erhobenen Daten konnte dies für die betrachteten Anweisungen grundsätzlich bestätigt werden. Daher wird abschließend gefolgert, dass die Aktionsausführung basierend auf Verbalisierten Physikalischen Effekten keine zur menschlichen Intuition gegensätzlichen Aktionsausführungen erzeugt und daher im Rahmen eines intuitiv zu verwendenden Gesamtsystem genutzt werden kann.

6.4 Zusammenfassung

Im Rahmen dieses Kapitels wurde die prototypische Umsetzung des Gesamtsystems vorgestellt, welches die in Kapitel 3, 4 und 5 vorgeschlagenen Konzepte und Verfahren realisiert. Hierzu wurden exemplarisch zwei Robotersysteme ausgewählt, welche sich aus verschiedenen Hard- und Softwarekomponenten zusammensetzen. Zur Umsetzung des Gesamtsystems wurden eine Reihe an Komponenten identifiziert, welche unabhängig vom Robotersystem beschrieben werden können und daher für beide Robotersysteme gemeinsam verwendet werden. Diese umfassen beispielsweise die Komponenten zur Verarbeitung von Eingaben gemäß der domänenspezifischen Sprache, die XML-basierte Aktionsbeschreibung, die Weltrepräsentation basierend auf dem ENACT-Framework sowie die grafische Benutzerschnittstelle.

Des Weiteren wurden eine Reihe an Komponenten beschrieben, welche individuell für ein entsprechendes Robotersystem umgesetzt wurden. Im Rahmen der prototypischen Umsetzung entsprechen diese zum einen den verwendeten Extraktionskomponenten, da beispielsweise verschiedene optische Sensoren oder Greifsysteme verwendet werden. Zum anderen betrifft dies die Einträge der Symbolinformation im Rahmen des physikalischen Wörterbuchs, welche die zur Extraktion verwendete Komponente und Sensorik spezifizieren.

Basierend auf der prototypischen Umsetzung wurde zunächst die flexible Verwendung der Aktionsbeschreibung evaluiert. Da die Aktionsdefinition in Form von Verbalisierten Physikalischen Effekten eine Aufgabenraum-basierte Aktionsbeschreibung darstellt, erlaubt diese prinzipiell eine vom Robotersystem unabhängige Aktionsdefinition. Hierzu wurde analysiert, welche Eigenschaften der Umgebung variiert werden können, ohne eine Änderung einer auszuführenden Anweisung zu erfordern. An dieser Stelle wurde die Variation des Robotersystems, der Roboterumgebung sowie der Objekteigenschaften identifiziert. Anhand eines exemplarischen Anwendungsszenarios wurde gezeigt, dass die Aktionsbeschreibung basierend auf Verbalisierten Physikalischen Effekten eine entsprechende Variation der Umgebungseigenschaften erlaubt und somit eine flexible Verwendung der beschriebenen Roboterfähigkeiten ermöglicht.

Ein weiterer Teil der Evaluation untersuchte die im Rahmen der Aufgabenstellung beschriebene wissenschaftliche Fragestellung **F4**, welche die intuitive Nutzung der vorgeschlagenen Konzepte und Verfahren in Form einer symbolischen Nutzerschnittstelle beschreibt. Diese lautet wie folgt:

F4 In wie weit kann basierend auf den vorgeschlagenen Konzepten ein intuitiv zu verwendendes Gesamtsystem umgesetzt werden?

Im Rahmen von Kapitel 5 wurde hierzu die Nutzung einer domänenspezifischen Sprache vorgeschlagen, welche die verbalen Ausdrücke eines Verbalisierten Physikalischen Effekts als Schlüsselworte und die obligatorischen Ergänzungen in einer geeigneten Ausdrucksform als Parameterphrasen verwendet. Zur Evaluation der intuitiven Verwendbarkeit wurden im Rahmen einer Benutzerevaluation empirische Daten erhoben und ausgewertet. Es wurden insgesamt Daten von 32 Personen ausgewertet, welche in vier Gruppe kategorisiert wurden. Das Spektrum reichte hierbei vom Standardbenutzer mit Allgemeinwissen bis hin zum Robotikexperten.

Die Benutzerevaluation wurde in zwei Bereiche unterteilt, welche zum einen die intuitive Nutzung der Eingabe in Form der domänenspezifischen Sprache und zum anderen die Ausführung einer instruierten Anweisung basierend auf Verbalisierten Physikalischen Effekten untersuchte.

Zur Evaluation der intuitiven Nutzung der Eingabe in Form der domänenspezifischen Sprache wurden eine Reihe an Aufgaben aus dem Bereich der Objektmanipulation spezifiziert. Bei der Durchführung der Evaluation wurde den Probanden zunächst eine entsprechende Aktionsausführung vorgeführt. Im Anschluss wurden die Probanden nach einer Beschreibung der vorgeführten Demonstration gefragt, welche in Form einer Instruktion, beziehungsweise in Form eines Imperativsatzes formuliert werden sollte. Auf Basis der betrachteten Aufgaben konnte gezeigt werden, dass die durch die Probanden spezifizierte Anweisungsstruktur im Allgemeinen der im Rahmen der domänenspezifischen Sprache verwendeten Abstraktion entspricht. Dadurch wurde abschließend gefolgert, dass die vorgeschlagene domänenspezifische Sprache eine intuitiv zu verwendete, symbolische Benutzerschnittstelle beschreibt.

Zur Evaluation der intuitiven Ausführung einer instruierten Anweisung basierend auf Verbalisierten Physikalischen Effekten wurden zunächst eine Reihe an Anweisungen gemäß der domänenspezifischen Sprache formuliert und den Probanden vorgelegt. Im Anschluss sollten die Probanden die zur Anweisung erwartete Ausführung aufzeigen, beziehungsweise unter Nutzung eines Robotersystems demonstrieren. Die demonstrierten Ausführungen wurden abschließend mit der Aktionsausführung basierend auf Verbalisierten Physikalischen Effekten verglichen und bewertet. Auf Basis der erhobenen empirischen Daten konnte gezeigt werden, dass die Ausführung einer Aktion basierend auf Verbalisierten Physikalischen Effekten im Allgemeinen der Ausführung entspricht, welche ein Benutzer vom System erwartet. Daher wurde abschließend gefolgert, dass die Aktionsausführung basierend auf Verbalisierten Physikalischen Effekten keine zur menschlichen Intuition gegensätzlichen Aktionsausführungen erzeugt und daher im Rahmen eines intuitiv zu verwendenden Gesamtsystem genutzt werden kann.

7 Schlussbemerkungen

Zum Abschluss dieser Arbeit werden die im Rahmen der einzelnen Kapitel beschriebenen, umgesetzten und evaluierten Konzepte, Verfahren und Erkenntnisse zusammengefasst und entsprechende Schlussfolgerungen abgeleitet (siehe Kapitel 7.1) sowie ein Ausblick über potentielle Erweiterungen präsentiert (siehe Kapitel 7.2).

7.1 Fazit

Im Rahmen der Motivation (siehe Kapitel 1.1) und Aufgabenstellung (siehe Kapitel 1.2) wurde beschrieben, dass die Einsatzgebiete von zukünftigen Robotersystemen auf neue Bereiche wie beispielsweise private Haushalte erweitert werden sollen.

Dies hat zur Folge, dass zukünftige Robotersysteme typischerweise nicht durch einen Experten, sondern durch eine Person ohne Expertenwissen im Bereich der Robotik verwendet werden. Daher muss davon ausgegangen werden, dass eine solche Person ausschließlich auf die Ausdrucksmöglichkeiten zurückgreifen kann, welche auch im Bereich der zwischenmenschlichen Kommunikation verwendet werden. Betrachtet man die typische Kommunikation des Menschen, so basiert diese im Allgemeinen auf symbolischer Information. Die verwendeten Symbole entsprechen beispielsweise Worte oder Gesten. Die Kommunikation eines Robotersystems basiert hingegen im Allgemeinen auf subsymbolischer Information, dargestellt durch einen abstrakten Befehlssatz mit konkreten Parameterwerten für beispielsweise Arbeitsraumpositionen, Regelsollwerte oder Regelstrategien.

Basierend darauf wurde gefolgert, dass ein zukünftiges Robotersystem eine intuitiv zu verwendende, symbolische Benutzerschnittstelle zur Verfügung stellen sollte. Zudem sollte das Robotersystem in der Lage sein, die symbolischen Benutzereingaben in eine zur Ausführung geeignete, subsymbolische Repräsentation zu überführen. Da diese Umwandlung im Allgemeinen nicht direkt durchgeführt werden kann, wird eine Transformationsschicht benötigt, welche die symbolischen Benutzereingaben in subsymbolische Roboteranweisungen überführt.

Ausgehend von einer dreischichtigen Systemarchitektur, welche sich aus der symbolischen Benutzerschicht, der im Rahmen dieser Arbeit zu spezifizierenden Transformationsschicht und der subsymbolischen Kontrollschicht zusammensetzt, wurde zunächst untersucht, in welcher Form die Schnittstellen zwischen den einzelnen Schichten umgesetzt werden können.

Zur Umsetzung der Schnittstelle zwischen der symbolischen Benutzerschicht und der Transformationsschicht wurde festgelegt, diese auf Basis von natürlichsprachlichen Symbolen in einer textuellen, semi-strukturierten Form zu spezifizieren. Zur Umsetzung der Schnittstelle zwischen der Transformationsschicht und der subsymbolischen Kontrollschicht wurde die Verwendung von Aktionsprimitivnetzen festgelegt.

Das Ziel dieser Arbeit bestand somit darin, eine Transformationsschicht zu realisieren, welche eine symbolische Anweisung basierend auf natürlichsprachlichen Symbolen entgegennimmt und diese in eine subsymbolische Repräsentation basierend auf Aktionsprimitivnetzen überführt. Dies umfasste die Identifikation der benötigten Komponenten sowie die Beschreibung, Umsetzung und Evaluation von geeigneten Lösungsansätzen. Hierzu wurden eine Reihe an wissenschaftlichen Fragestellungen **F1** - **F4** formuliert und untersucht. Die im Rahmen der jeweiligen Bearbeitung vorgeschlagenen Konzepte und die daraus gewonnenen Erkenntnisse werden im Folgenden zusammengefasst.

F1 In welcher Form kann die Semantik einer ausführbaren Aktion systematisch beschrieben und in einem Robotersystem hinterlegt werden?

Bevor ein Konzept zur systematischen Beschreibung der Semantik von ausführbaren Roboteraktionen vorgeschlagen werden konnte, musste zunächst eine geeignete Zwischenrepräsentation gefunden werden, welche sowohl für die symbolische Darstellung des Menschen als auch für die subsymbolische Darstellung des Robotersystems gültig ist und als Basis der Beschreibung der Zusammenhänge zwischen symbolischer und subsymbolischer Information dienen konnte. Hierzu wurde als Grundansatz eine Symbolbeschreibung auf Basis von physikalischen Effekten, beziehungsweise physikalischen Gesetzmäßigkeiten, Größen und Nebenbedingungen vorgeschlagen.

Zur systematischen Beschreibung der Semantik von ausführbaren Aktionen wurde zunächst das Konzept der Prinzipiellösungen aus dem Bereich der Systementwicklung für die Verwendung von physikalischen Größen adaptiert. Daraus resultieren eine Reihe von physikalischen Operationen auf physikalischen Größen. Basierend auf diesen wurde das Konzept der *Prinzipiellen Physikalischen Effekte* eingeführt, welches die Zusammenhänge zwischen der gewählten Zwischenrepräsentation und der subsymbolischen Repräsentation des Robotersystems beschreibt. Aufbauend darauf wurde das Konzept der *Verbalisierten Physikalischen Effekte* vorgeschlagen, welches die Zusammenhänge zwischen einer symbolischen Beschreibung in Form eines Verbs und einem Prinzipiellen Physikalischen Effekt beschreibt.

- F2** In welcher Form kann die Semantik der zur Parametrisierung einer ausführbaren Aktion verwendeten Symbole systematisch beschrieben und in einem Robotersystem hinterlegt werden?

Aufbauend auf der vorgeschlagenen Aktionsbeschreibung wurde identifiziert, dass im Rahmen einer symbolischen Anweisung im Allgemeinen verschiedene Arten von Symbolen verwendet werden. Neben dem Verb, welches typischerweise die auszuführende Aktion kennzeichnet sind dies Symbole, welche die obligatorischen und fakultativen Ergänzungen des verwendeten Verbs darstellen und die auszuführende Aktion genauer spezifizieren. Um eine Kompatibilität mit der vorgeschlagenen Aktionsbeschreibung zu gewährleisten, mussten diese Symbole ebenfalls auf Basis von physikalischen Eigenschaften beschrieben werden.

Hierzu wurde zunächst analysiert, welche Arten von Symbolen im Rahmen von symbolischen Anweisungen verwendet werden, sowie in welcher Form diese eine auszuführende Aktion beeinflussen. Basierend auf einer Analyse der Symbole wurden eine Reihe an Eigenschaften extrahiert, welche im Rahmen eines *physikalischen Wörterbuchs* beschrieben und im System hinterlegt werden. Diese Komponente stellt einen Teil der Wissensbasis des Robotersystems dar und beschreibt für jedes zur Verfügung stehende Symbol die Symbolsemantik im Kontext von physikalischen Eigenschaften. Zudem enthält das physikalische Wörterbuch die Information darüber, mit welcher im System zur Verfügung stehenden Sensorik und Extraktionskomponenten eine physikalische Eigenschaft durch das Robotersystem zur Laufzeit extrahiert werden kann.

- F3** In welcher Form können die vorgeschlagenen Konzepte zur Beschreibung der Symbolsemantik gemeinsam verwendet werden um eine subsymbolische Parametrierung der auszuführenden Roboteraktionen zu ermöglichen?

Zur gemeinsamen und intuitiven Nutzung der eingeführten Konzepte in Form einer symbolischen Benutzerschnittstelle wurde die Verwendung im Rahmen einer *domänenspezifischen Sprache* vorgeschlagen. Diese nutzt die zu den auszuführenden Verbalisierten Physikalischen Effekten zugeordneten Verben als Schlüsselworte und die entsprechenden obligatorischen Ergänzungen in einer geeigneten Ausdrucksformen als Parameterphrasen. Zur Definition von validen Anweisungen werden die ausführbaren Aktionen auf Basis einer XML-basierten Beschreibung im System hinterlegt, welche gemäß eines XML-Schemas spezifiziert werden. Diese Form der Aktionsbeschreibung wird unabhängig vom Robotersystem spezifiziert, wodurch diese flexibel für verschiedene Robotersysteme genutzt werden kann.

Zur Parametrisierung der auszuführenden Roboteraktionen basierend auf den symbolisch instruierten Anweisungen wurden zwei Verfahren vorgestellt, welche zum einen das Auflösen von symbolischer Information und zum anderen die Extraktion von subsymbolischer Information als auch die Parametrisierung des auszuführenden Aktionsprimitivnetz beschreiben.

Das Verfahren zum Auflösen der symbolischen Information nutzt die im Robotersystem hinterlegte Symbolinformation, welche im Rahmen des physikalischen Wörterbuchs und der XML-basierten Aktionsbeschreibung zur Verfügung steht. Das Verfahren zur Extraktion und zur Parametrisierung des auszuführenden Aktionsprimitivnetzes nutzt insbesondere die zur Verfügung stehenden Extraktionskomponenten zur Bestimmung der physikalischen Eigenschaften von realen Objekten der Roboterumgebung.

F4 In wie weit kann basierend auf den vorgeschlagenen Konzepten ein intuitiv zu verwendendes Gesamtsystem umgesetzt werden?

Hierzu wurden zunächst die vorgeschlagenen Konzepte und Verfahren im Rahmen eines prototypischen Gesamtsystems umgesetzt. Basierend auf diesem wurde eine Benutzerevaluation durchgeführt, welche die Untersuchung der intuitiven Verwendung des Gesamtsystems zum Ziel hatte. Hierbei wurden empirische Daten von insgesamt 32 Probanden erhoben und analysiert. Das Spektrum der Probanden reichte vom Benutzer ohne Expertenwissen im Bereich der Robotik bis hin zum Robotikexperten.

Die Durchführung der Benutzerevaluation wurde in zwei Bereiche unterteilt, welche zum einen die intuitive Nutzung der Eingabe in Form der domänenspezifischen Sprache und zum anderen die Ausführung einer instruierten Anweisung basierend auf Verbalisierten Physikalischen Effekten untersuchte. Zur Evaluation der intuitiven Nutzung der Eingabe in Form der domänenspezifischen Sprache wurden eine Reihe an Aufgaben vorgegeben und die Probanden mussten entsprechende Instruktionen zur Ausführung durch das Robotersystem nennen. Zur Evaluation der intuitiven Ausführung einer instruierten Anweisung basierend auf Verbalisierten Physikalischen Effekten wurden eine Reihe an Anweisungen vorgegeben, für diese die Probanden die erwartete Ausführung durch das Robotersystem aufzeigen sollten. Diese wurde unter Nutzung eines Robotersystems demonstriert und auf Basis der Abweichungen zwischen der Demonstration und der Ausführung basierend auf dem Verbalisierten Physikalischen Effekt bewertet. Anhand der erhobenen empirischen Daten konnte gezeigt werden, dass auf Basis der vorgeschlagenen Konzepte ein intuitiv zu verwendendes Gesamtsystem umgesetzt werden kann.

Abschließend lassen sich die im Rahmen dieser Arbeit beschriebenen Konzepte und Verfahren, die Verarbeitung einer symbolischen Anweisung in Form der domänenspezifischen Sprache, sowie die im System vorhandenen Komponenten in Form eines Schaubilds zusammenfassen (siehe Abbildung 7.1).

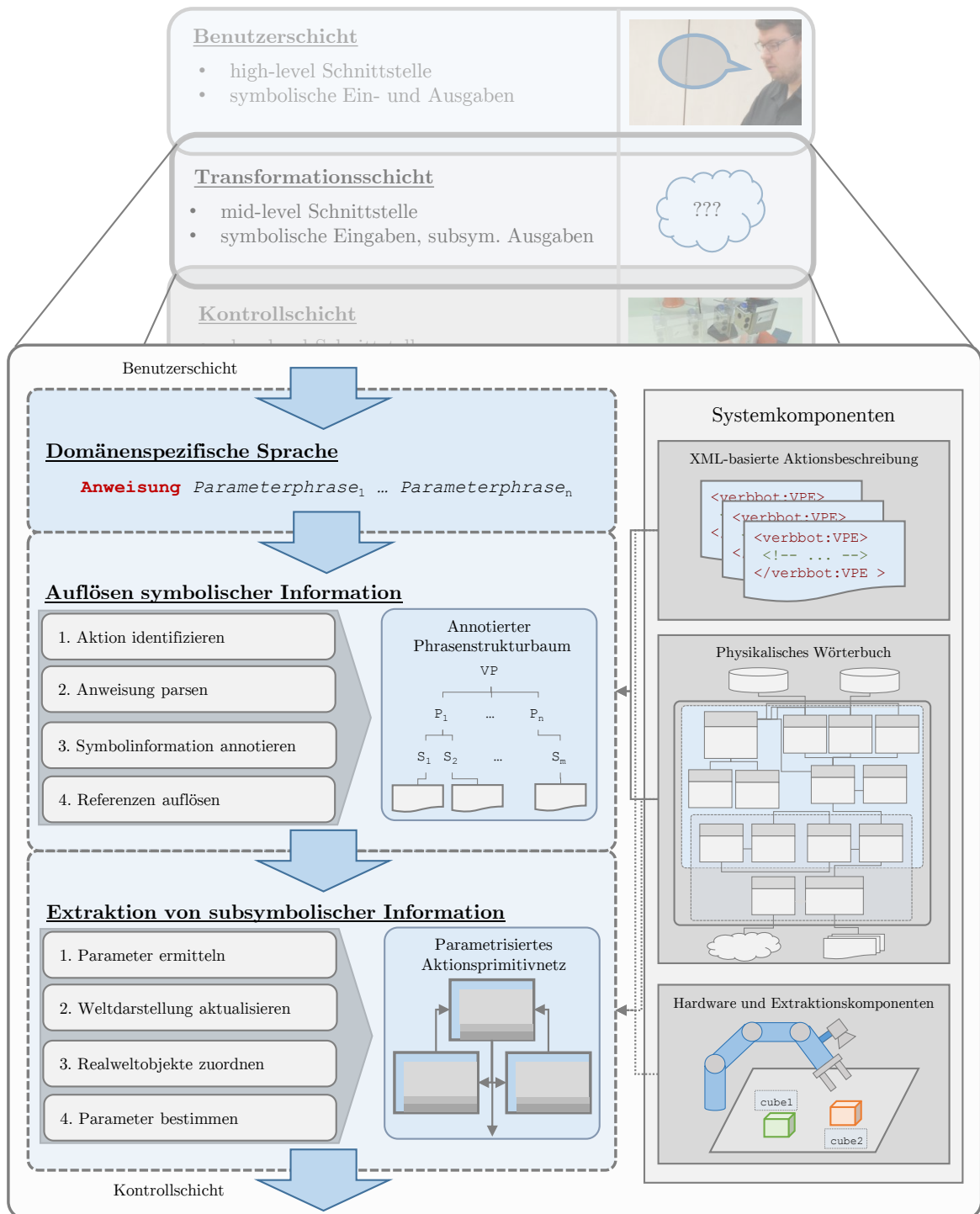


Abbildung 7.1: Darstellung der im Rahmen dieser Arbeit vorgeschlagenen Komponenten zur Umsetzung einer Transformationsschicht, welche symbolische Eingaben in Form einer domänenspezifischen Sprache entgegennimmt und diese unter Nutzung von Systemkomponenten in eine zur Ausführung geeignete, subsymbolische Repräsentation basierend auf Aktionsprimitivnetzen überführt.

7.2 Ausblick

Die vorgeschlagenen Konzepte und Verfahren zeigten bereits gute Ergebnisse, beispielsweise im Bereich der flexiblen Verwendung der auf Basis von Verbalisierten Physikalischen Effekten spezifizierten Aktionen oder der intuitiven Verwendung in Form der domänenspezifischen Sprache. Dennoch existieren verschiedene Möglichkeiten und Potentiale zur Erweiterung des Gesamtsystems, welche im Folgenden genannt werden.

Die möglichen Erweiterungen werden hierbei in vier Kategorien gegliedert, diese lauten wie folgt:

- Erweiterung im Bereich der *Aktionsbeschreibung*
- Erweiterung im Bereich der *Benutzerschnittstelle*
- Erweiterung im Bereich der *Systemkomponenten*
- Erweiterung im Bereich der *Anwendungsgebiete*

Die Erweiterungen im Bereich der *Aktionsbeschreibung* umfassen beispielsweise die Untersuchung der im Rahmen dieser Arbeit nicht betrachteten komplexen und zusammengesetzte physikalische Effekte. Für die komplexen physikalischen Effekte ist zu untersuchen, in welcher Form, beziehungsweise durch welche Verben diese im Rahmen einer symbolischen Anweisung verwendet werden. Für die zusammengesetzten physikalischen Effekte ist beispielsweise zu untersuchen, in welcher Form diese aus elementaren und komplexen physikalischen Effekten zusammengesetzt werden können und in welcher Form diese Verkettung von Effekten in einer symbolischen Anweisung berücksichtigt wird.

Die Erweiterungen im Bereich der *Benutzerschnittstelle* umfassen beispielsweise die Erweiterung der domänenspezifischen Sprache dahingehend, dass ein Benutzer zusätzlich zu den obligatorischen Ergänzungen weitere fakultative Ergänzungen im Rahmen einer symbolischen Anweisung nutzen kann. Hier ist zu untersuchen, in welcher Form fakultative Ergänzungen zu einem verbalen Ausdruck möglich sind sowie in welcher Form diese die Parametrisierung der auszuführenden Aktion beeinflussen. Eine weitere Möglichkeit stellt die Erweiterung durch eine physikalische Simulation dar, welche beispielsweise als Nutzerfeedback verwendet werden kann. Mit dieser kann eine Aktion vor der Ausführung dem Benutzer in einer virtuellen Umgebung gezeigt und beispielsweise eine fehlerhafte Ausführung vermieden werden. Neben der Visualisierung für den Benutzer kann diese auch intern vom Robotersystem genutzt werden, um beispielsweise die ermittelte Parametrisierung zu verifizieren. Eine weitere Möglichkeit zur Erweiterung ist beispielsweise die Verwendung eines Dialogsystems. Dies kann zum einen dazu genutzt werden, Konflikte bei der ermittelten Parametrisierung oder der Aufgabenausführung zu lösen. Zum anderen könnte dies genutzt werden, um neue Einträge für das physikalische Wörterbuch dialogbasiert zu lernen. Damit verbunden ist die allgemeine Erweiterung um zusätzliche Eingabemodalitäten wie beispielsweise Sprache oder Gesten.

Die Erweiterungen im Bereich der *Systemkomponenten* umfassen beispielsweise die Erweiterung um eine Komponente zur Aufgabenkoordination. Diese wird benötigt, da beispielsweise eine symbolische Anweisung eine Mehrfachausführung erfordert (z.B. Greife *alle* Objekte!), was im Rahmen dieser Arbeit nicht betrachtet wurde. Das Spektrum einer solchen Komponente reicht von der Koordination einer Mehrfachausführung bis zur kollaborativen Ausführung von instruierten Aufgaben. Eine weitere Möglichkeit stellt die Erweiterung um zusätzliche Extraktionskomponenten dar, welche die Qualität der zur Ausführung bestimmten Parameter beeinflussen.

Die Erweiterungen im Bereich der *Anwendungsgebiete* umfassen beispielsweise die Nutzung der vorgeschlagenen Aktionsbeschreibung im Rahmen eines symbolischen Planungssystems. Hierbei stellen die auf Basis von Verbalisierten Physikalischen Effekten beschriebenen Roboteraktionen flexible Roboterfähigkeiten zur Verfügung, welche zur Ausführung für verschiedene Roboterkonfigurationen genutzt werden können. Ein weiterer Anwendungsbereich stellt beispielsweise die Nutzung im Rahmen eines Programmieren durch Vormachen (PdV)-Systems dar. Hier können die Verbalisierten Physikalischen Effekte im Rahmen der Demonstrationsphase identifiziert werden um eine symbolische Beschreibung oder die sensorbasierte Ausführung einer Demonstration zu extrahieren.

Abbildungsverzeichnis

1.1	Übersicht der Fähigkeiten eines Robotersystems, welche zur Erweiterung der Einsatzgebiete für zukünftige Robotersysteme benötigt werden (nach [SPARC15]).	2
1.2	Beispiele für Servicerobotersysteme, welche für den Einsatz in Umgebungen wie privaten Haushalten konzipiert werden und über fortschrittliche Fähigkeiten in verschiedenen Bereichen wie beispielsweise der Interaktion, der Kognition oder der Handhabung verfügen müssen (Bildquellen: links [TUM16], rechts [IPA16]).	3
1.3	Grafische Darstellung der im Rahmen dieser Arbeit untersuchte Problemstellung in Form einer 3-Schicht-Architektur. Da im Allgemeinen keine direkte Abbildung zwischen symbolischer und subsymbolischer Information möglich ist, wird eine Transformationsschicht zwischen den beiden Repräsentationen benötigt (grün). Die konzeptionelle Beschreibung einer solchen Transformationsschicht nimmt eine zentrale Stellung in dieser Arbeit ein.	4
1.4	Übersicht der Fähigkeiten eines Robotersystems, deren Komplexität im Rahmen dieser Arbeit durch eigene wissenschaftliche Beiträge weiterentwickelt werden (grün).	6
1.5	Grafische Darstellung der im Rahmen dieser Arbeit umzusetzenden Mensch-Roboter-Interaktion. Die Merkmale zur Beschreibung einer Mensch-Roboter-Interaktion sind aus [Onnasch16] entnommen. Die verwendeten Merkmale werden kategorisiert in Merkmale zur Interaktions- (erste Spalte), Team- (zweite und dritte Spalte) und Roboterklassifikation (vierte und fünfte Spalte).	7
1.6	Grafische Darstellung der im Rahmen dieser Arbeit umzusetzenden kognitiven Fähigkeiten. Die Merkmale zur Beschreibung sind angelehnt an [SPARC15] und werden unterteilt in Merkmale zur Ein- und Ausgabeklassifikation (erste Spalte) sowie der Systemklassifikation (zweite und dritte Spalte).	9
1.7	Schematische Zuordnung der einzelnen Kapitel dieser Arbeit.	11
2.1	Inhaltliche Zuordnung des Kapitels im Gesamtsystem.	14
2.2	Beispiel zur Verdeutlichung der Unterschiede zwischen den verschiedenen Formen (strukturiert, semi-strukturiert und unstrukturiert) einer textuellen Repräsentation.	16
2.3	Inhaltliche Zuordnung des Kapitels im Gesamtsystem.	18
2.4	Beispiele zur Definition von sensorbasierten Roboteraktionen gemäß des <i>Task Frame Formalismus</i> (TFF) nach [Bruyninckx96]. Links ist die Definition einer sensorüberwachten Bewegung, rechts die Definition einer sensorgeführten Bewegung dargestellt.	19
2.5	Inhaltliche Zuordnung des Kapitels im Gesamtsystem.	23

2.6	Prinzipielle Komponenten eines Robotersystems, welches die Verarbeitung von strukturierten Eingaben und Ausführung von Grobbewegungen erlaubt.	25
2.7	Prinzipielle Komponenten eines Robotersystems, welches die Verarbeitung von strukturierten Eingaben und Ausführung von Feinbewegungen erlaubt.	25
2.8	Prinzipielle Komponenten eines Robotersystems, welches die Verarbeitung von semi-strukturierten Eingaben und Ausführung von Grobbewegungen erlaubt. . .	26
2.9	Prinzipielle Komponenten eines Robotersystems, welches die Verarbeitung von semi-strukturierten Eingaben und Ausführung von Feinbewegungen erlaubt. . . .	26
2.10	Prinzipielle Komponenten der Transformationsschicht.	28
2.11	Anwendungsgebiete zukünftiger Robotersysteme, bei denen sensorbasierte Bewegungen vorteilhaft eingesetzt werden können. Links eine Anwendung aus dem Bereich Catering, rechts eine Anwendung aus dem Bereich Werkstatt.	36
3.1	Beispiel für eine Aufgabe aus dem Bereich der Objektmanipulation inklusive der Unterteilung der Ausführungsphasen in die Annäherungs-, Manipulations- und Abschlussphase.	49
3.2	Allgemeine Struktur des zu einem Prinzipiellen Physikalischen Effekt zugeordneten Aktionsprimitivnetz.	51
3.3	Beispiel für die Abhängigkeit des Aktionsprimitivnetzes von der im Robotersystem verfügbaren Sensorik. Im linken Teil der Abbildung ist eine exemplarische Szene sowie das zur Aktionsausführung benötigte Aktionsprimitivnetz für die Nutzung von globaler Sensorik dargestellt. Im rechten Teil der Abbildung ist eine exemplarische Szene sowie das zur Aktionsausführung unter Nutzung lokaler Sensorik erweiterte Aktionsprimitivnetz dargestellt.	52
3.4	Möglichkeiten zur Festlegung der Position (einer Koordinate) und Orientierung (einer Achse) des lokalen Koordinatensystems (<i>task frame</i>) \mathcal{TF} einer auszuführenden Aktion.	53
3.5	Beispiele für Probleme bei Verwendung eines lokalen Koordinatensystems, falls bei Ausführung einer Anweisung nicht ausschließlich ein Kontaktpunkt vorhanden ist.	54
3.6	Beispiele der Ausführungen von PPE, welche den Probanden im Rahmen der Benutzerstudie vorgeführt werden. Links oben (Aufgabe 1) zeigt die Ausführung des Effekts <i>Transformiere eine Kraft in eine Beschleunigung, bzw. Länge</i> . Rechts oben (Aufgabe 2) zeigt die Ausführung des Effekts <i>Ändere die potentielle Energie</i> . Links unten (Aufgabe 3) zeigt die Ausführung des Effekts <i>Transformiere einen Impuls in eine Geschwindigkeit, bzw. Länge</i> . Rechts unten (Aufgabe 4) zeigt die Ausführung des Effekts <i>Absorbiere eine Kraft</i>	61
3.7	Verfügbarer Befehlssatz für den zweiten Teil der Benutzerstudie.	61
3.8	Schematische Zusammenhänge der im Rahmen dieses Kapitels erarbeiteten und beschriebenen Konzepte.	66
4.1	Exemplarische Annotation der Struktur einer Anweisung. Die Ausgabe wird durch einen Parser nach [Klein03] beziehungsweise unter Berücksichtigung der universellen Beschreibung nach [Petrov11] erzeugt.	69

4.2	Einfluss des Artikels auf die Ausführung einer Beispielanweisung.	72
4.3	Schematische Darstellung der im Rahmen des physikalischen Wörterbuchs hinterlegten Information (weiße Umrandung). Dies umfasst zum einen Information zur Beschreibung der Symbolsemantik in allgemeiner Form sowie im Kontext der physikalischen Eigenschaften (blaue Umrandung). Zum anderen umfasst dies Information zur automatischen Parametrisierung und Extraktion von physikalischen Eigenschaften (graue Umrandung). Zudem sind eine Reihe externer Komponenten vorhanden. Dies sind beispielsweise die Sensorik des Robotersystems oder die Komponenten zur Extraktion der physikalischen Eigenschaften.	75
4.4	Exemplarische Einträge des physikalischen Wörterbuchs.	76
4.5	Exemplarische Einträge des physikalischen Wörterbuchs.	77
5.1	Kategorisierung der Einsatzgebiete und Einordnung der betrachteten Arbeiten, welche domänenspezifische Sprachen im Bereich der Robotik verwenden. Die Kategorien und Einordnung sind aus [Nordmann14] entnommen.	80
5.2	Resultierender Phrasenstrukturbaum für eine Beispielanweisung.	90
5.3	Ausschnitt der Annotation des Symbols <code>cube</code> mit Information aus dem physikalischen Wörterbuch.	91
5.4	Einschränkung der physikalischen Eigenschaftswerte des Symbols <code>cube</code> aufgrund des im Rahmen der symbolischen Anweisung vorhandenen Parametersymbols <code>red</code>	91
5.5	Erfassung und Annotation der in der Roboterumgebung vorhandenen Objekte. Im linken Bild ist eine Übersicht der Szene inklusive des zur Erfassung der Objekte verwendeten optischen Sensors (siehe grüne Umrandung) abgebildet. Im mittleren Bild ist die Szene aus Sicht des Robotersystems dargestellt. Im rechten Bild sind die auf Basis der im System verfügbaren Extraktionskomponente erkannten und annotierten Objekte illustriert.	95
5.6	Zuordnung des realen Objekts <code>cube</code> zu dem im Rahmen des annotierten Phrasenstrukturbaum spezifizierten Objekts (siehe Abbildung 5.4).	96
5.7	Schematische Zusammenhänge der im Rahmen dieses Kapitels beschriebenen Verfahren.	97
6.1	Illustration der zur prototypischen Umsetzung verwendeten Robotersysteme. Das System aus der linken Abbildung setzt sich aus einem KUKA LBR4, einem 2-Finger-Parallelgreifer PG70 von Schunk, sowie einer Microsoft Kinect zusammen. Das System aus der rechten Abbildung setzt sich aus einem KUKA LBR4+, einem adaptiven 3-Finger-Robotergriffe von Robotiq, einem Ensenso N10 Tiefensensor und einer IDS UI-1220SE Industriekamera zusammen.	100
6.2	Darstellung einer Beispielszene (links) und zugehörige Komponenten der Weltrepräsentation basierend auf dem ENACT Framework (rechts). Für jedes erkannte Objekt der Szene ist eine Entität mit einer Reihe von Aspekten vorhanden, welche über entsprechende Aktoren spezifiziert werden. Zudem ist typischerweise eine Entität vorhanden, welche das Robotersystem beschreibt.	101
6.3	Grafische Benutzeroberfläche zur Eingabe von auszuführenden Anweisungen, zum Laden der Wissensbasis und zur Visualisierung der einzelnen Verarbeitungsschritte.	102

6.4	Beispiel für die Anpassung der im physikalischen Wörterbuch hinterlegten Symbolinformation zur Extraktion einer physikalischen Eigenschaft für verschiedene Robotersysteme. Die allgemeine Beschreibung der Symbolsemantik ist unabhängig vom Robotersystem spezifiziert und kann daher allgemein verwendet werden (grau). Die zur Extraktion verwendete Sensorik muss an entsprechender Stelle spezifiziert werden (blaue Umrandung).	103
6.5	Beispiele für Variationen der Roboterumgebung, welche auf Basis der Spezifikation von sensorbasierten Roboteraktionen kompensiert werden können.	104
6.6	Beispiele für Variationen der zu manipulierenden Objekte, beziehungsweise Objekteigenschaften, welche auf Basis der Extraktionskomponenten kompensiert werden können.	105
6.7	Stammdaten der im Rahmen der Evaluation verwendeten Objektsätze.	107
6.8	Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System1, Umgebung1 und Objektsatz1	108
6.9	Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System1, Umgebung1 und Objektsatz2	108
6.10	Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System1, Umgebung2 und Objektsatz1	109
6.11	Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System1, Umgebung2 und Objektsatz2	109
6.12	Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System2, Umgebung1 und Objektsatz1	110
6.13	Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System2, Umgebung1 und Objektsatz2	110
6.14	Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System2, Umgebung2 und Objektsatz1	111
6.15	Extrahierte Parameter und Ausführung des exemplarisch gewählten Anwendungsszenarios bei Verwendung der Umgebungsvariante System2, Umgebung2 und Objektsatz2	111
6.16	Visualisierung der unterschiedlichen Wegstrecken unter Nutzung der Umgebungsvarianten Umgebung1 und Umgebung2 , exemplarisch dargestellt für die weiteren Parameter System1 und Objektsatz1	112
6.17	Visualisierung der Farbbestimmung für unterschiedliche Beleuchtungsverhältnisse unter Nutzung der Umgebungsvarianten Umgebung1, System1 und Objektsatz1	113

6.18	Im Rahmen von Teil B der Benutzerevaluation verwendete Umgebung (links) und auszuführende Anweisungen gemäß der domänenspezifischen Sprache (rechts).	121
6.19	Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung <i>berühre den roten Würfel</i> .	122
6.20	Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung <i>schiebe den roten Würfel gegen den grauen Würfel</i> .	123
6.21	Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung <i>drehe den roten Würfel um 90 Grad nach rechts</i> . Zusätzlich dargestellt ist die während der Manipulationsphase relevante rotatorische Bewegung. Die dargestellten Zeiten t_{start} und t_{end} entsprechen den normierten Start- und Endzeitpunkten der Manipulationsphase.	123
6.22	Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung <i>kippe den roten Würfel um 45 Grad nach links</i> . Zusätzlich dargestellt ist die während der Manipulationsphase relevante rotatorische Bewegung. Die dargestellten Zeiten t_{start} und t_{end} entsprechen den normierten Start- und Endzeitpunkten der Manipulationsphase.	124
6.23	Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung <i>hebe den roten Würfel um 30 cm</i> .	125
6.24	Exemplarische Demonstration eines Probanden sowie Abweichungen der demonstrierten Bahn gegenüber der Ausführung basierend auf Verbalisierten Physikalischen Effekten für die Anweisung <i>senke den roten Würfel um 30 cm</i> .	125
7.1	Darstellung der im Rahmen dieser Arbeit vorgeschlagenen Komponenten zur Umsetzung einer Transformationsschicht, welche symbolische Eingaben in Form einer domänenspezifischen Sprache entgegennimmt und diese unter Nutzung von Systemkomponenten in eine zur Ausführung geeignete, subsymbolische Repräsentation basierend auf Aktionsprimitivnetzen überführt.	133

Tabellenverzeichnis

2.1	Sensorische Modalitäten des Menschen gemäß [Luczak93] bzw. [Roetting05]. Die blau hervorgehobenen Einträge stellen die Modalitäten dar, welche typischerweise zur Benutzerinteraktion im Rahmen eines technischen Systems verwendet werden.	15
2.2	Klassifikation der eigenen und verwandten Arbeiten auf Basis der umgesetzten Benutzer- und Kontrollschicht. Für die einzelnen Gesamtsysteme wurde jeweils eine stellvertretende Publikation ausgewählt.	24
2.3	Kategorisierung der eigenen und verwandten Arbeiten gemäß der festgelegten Merkmale. Die Beschreibung der Symbole ist der dargestellten Legende zu entnehmen.	33
3.1	Liste der elementaren Operationen auf Energien, beziehungsweise Energiekomponenten nach [Koller98].	40
3.2	Liste der physikalischen Basisgrößen nach [ISO09].	40
3.3	Liste der relevanten geometrischen (graue Hervorhebung), kinematischen (grau-blaue Hervorhebung) und dynamischen (blaue Hervorhebung) Größen, welche im Rahmen von mechanischen Vorgängen verwendet werden (nach [Awrejcewicz12]).	41
3.4	Liste der <i>Physikalischen Operationen</i> . Diese werden unterteilt in <i>elementare Operationen</i> (<i>absorb</i> , <i>change</i> , <i>transform</i>) und <i>komplexe Operationen</i> (<i>merge</i> , <i>split</i>). .	42
3.5	Zusammenhänge zwischen den elementaren physikalischen Effekten (<i>absorb</i> , <i>change</i> , <i>transform</i>), der zugeordneten physikalischen Gesetzmäßigkeit und während der Effektausführung relevanten physikalischen Nebenbedingungen.	45
3.6	Übersicht der Ergänzungsklassen nach [Schumacher04].	58
3.7	Struktur, Satzbauplan und Ausdrucksformen für die Beispielanweisung “ <i>Schiebe den roten Block gegen den blauen Block!</i> “. Die Struktur und der Satzbauplan sind aus [IDS16] entnommen. Die Ausdrucksform wird auf Basis eines Parsers nach [Klein03] erzeugt.	59
3.8	Ergebnisse der Benutzerstudie.	62
3.9	Qualitative Zusammenhänge zwischen symbolischen Anweisungen, Satzbauplänen, Strukturen und Parameter des zu einem Verb zugeordneten PPE. Bei den verwendeten Beispielanweisungen handelt es sich um konkrete Anweisungen, welche im Rahmen der Benutzerstudie durch Probanden instruiert wurden.	64
6.1	Darstellung der im Rahmen der Benutzerevaluation verwendeten Aktionen inklusive der Verwendung gemäß der domänenspezifischen Sprache sowie dem zugeordneten Verbalisierten Physikalischen Effekt.	116

6.2	Darstellung der im Rahmen von Teil A der Benutzerevaluation ermittelten Daten basierend auf einer jeweiligen Demonstration der auszuführenden Aktion.	117
-----	------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----

Quellenverzeichnis

- [Aggarwal12] Aggarwal, Charu and Zhai, ChengXiang.
Mining text data.
 ISBN: 978-1-4614-3222-7.
 Springer Science & Business Media, 2012.
- [Apple16] Apple Inc.
Digitaler Assistent Siri.
<http://www.apple.com/de/ios/siri/>
 (Letzter Aufruf: 16.11.2016). 2016.
- [Argall09] Argall, Brenna and Chernova, Sonia and Veloso, Manuela and
 Browning, Brett.
A survey of robot learning from demonstration.
 Robotics and Autonomous Systems (RAS).
 vol. 57, no. 5, pp. 469-483.
 Elsevier, 2009.
- [Argall10] Argall, Brenna and Billard, Aude.
A survey of tactile human-robot interactions.
 Robotics and Autonomous Systems (RAS).
 vol. 58, no. 10, pp. 1159-1176.
 Elsevier, 2010.
- [Awrejcewicz12] Awrejcewicz, Jan.
Classical mechanics: kinematics and statics.
 Springer Advances in Mechanics and Mathematics.
 Springer Science & Business Media, 2012.
- [Ballard84] Ballard, Dana.
Task Frames in Robot Manipulation.
 Association for the Advancement of Artificial Intelligence (AAAI)-84.
 vol. 19, pp. 16-22.
 AAAI, 1984.

- [Beetz11] Beetz, Michael and Klank, Ulrich and Kresse, Ingo and Maldonado, Alexis and Mösenlechner, Lorenz and Pangercic, Dejan and Rühr, Thomas and Tenorth, Moritz.
Robotic roommates making pancakes.
International Conference on Humanoid Robots.
pp. 529–536.
IEEE, 2011.
- [Benali04] Benali-Khoudja, Mohamed and Hafez, Moustapha and Alexandre, Jean-Marc and Kheddar, Abderrahmane.
Tactile interfaces: a state-of-the-art survey.
International Symposium on Robotics (ISR).
vol. 31, pp. 23-26.
Citeseer, 2004
- [Billard08] Billard, Aude and Calinon, Sylvain and Dillmann, Ruediger and Schaal, Stefan.
Robot programming by demonstration.
Springer Handbook of Robotics.
pp. 1371-1394.
Springer, 2008.
- [Bollini13] Bollini, Mario and Tellex, Stefanie and Thompson, Tyler and Roy, Nicholas and Rus, Daniela.
Interpreting and executing recipes with a cooking robot.
Experimental Robotics.
pp. 481-195.
Springer, 2013.
- [Brants02] Brants, Sabine and Dipper, Stefanie and Hansen, Silvia and Lezius, Wolfgang and Smith, George.
The TIGER treebank.
Proceedings of the workshop on treebanks and linguistic theories.
vol. 168, 2002.
- [Bruyninckx96] Bruyninckx, Herman and De Schutter, Joris.
Specification of force-controlled actions in the “task frame formalism”.
Transactions on Robotics and Automation.
vol. 12, no. 4, pp. 581-589.
IEEE, 1996.
- [Bruyninckx03] Bruyninckx, Herman and Soetens, Peter and Koninckx, Bob.
The real-time motion control core of the Orocos project.
International Conference on Robotics and Automation (ICRA).
vol. 2, pp. 2766-2771.
IEEE, 2003.

- [Buchmann13] Buchmann, Thomas and Baumgartl, Johannes and Henrich, Dominik and Westfechtel, Bernhard.
Towards A Domain-specific Language For Pick-And-Place Applications.
International Workshop on Domain-Specific Languages and Models for Robotic Systems (DSLRob'13).
2013.
- [Bussmann08] Bussmann, Hadumod and Lauffer, Hartmut.
Lexikon der Sprachwissenschaft.
ISBN: 978-3-520-45204-7.
Kröner Stuttgart, 2008.
- [Cangelosi06] Cangelosi, Angelo.
The grounding and sharing of symbols.
Pragmatics & Cognition.
vol. 14, no. 2, pp. 275-285.
John Benjamins Publishing Company, 2006.
- [Coradeschi13] Coradeschi, Silvia and Loutfi, Amy and Wrede, Britta.
A short review of symbol grounding in robotic and intelligent systems.
KI-Künstliche Intelligenz.
vol. 27, no. 2, pp. 129-136.
Springer, 2013.
- [Craig05] Craig, John.
Introduction to Robotics: Mechanics and Control.
ISBN: 0201543613.
Pearson Prentice Hall Upper Saddle River, 2005
- [DeSchutter88] De Schutter, Joris and Van Brussel, Hendrik.
Compliant robot motion I.
A formalism for specifying compliant motion tasks.
The International Journal of Robotics Research.
vol. 7, no. 4, pp. 3-17.
SAGE Publications, 1988.
- [DeSchutter07] De Schutter, Joris and De Laet, Tinne and Rutgeerts, Johan and Decré, Wilm and Smits, Ruben and Aertbeliën, Erwin and Claes, Kasper and Bruyninckx, Herman.
Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty.
The International Journal of Robotics Research.
vol. 26, no. 5, pp. 433-455.
SAGE Publications, 2007.

- [Dzifcak09] Dzifcak, Juraj and Scheutz, Matthias and Baral, Chitta and Schermerhorn, Paul.
What to do and how to do it: Translating natural language directives into temporal and dynamic logic representation for goal management and action execution.
International Conference on Robotics and Automation (ICRA).
pp. 4163-4168.
IEEE, 2009.
- [Fikes71] Fikes, Richard and Nilsson, Nils.
STRIPS: A new approach to the application of theorem proving to problem solving.
Artificial intelligence.
vol. 2, no. 3-4, pp. 189-208.
Elsevier, 1971.
- [Finkemeyer04] Finkemeyer, Bernd.
Robotersteuerungsarchitektur auf der Basis von Aktionsprimitiven.
ISBN: 3832228934.
Shaker Verlag, 2004.
- [Flanagan06] Flanagan, Randall and Bowman, Miles and Johansson, Roland.
Control strategies in object manipulation tasks.
Current opinion in neurobiology.
vol. 16, no. 6, pp. 650-659.
Elsevier, 2006.
- [Gavrila99] Gavrila, Dariu.
The visual analysis of human movement: A survey.
Computer Vision and Image Understanding.
vol. 73, no. 1, pp. 82-98.
Elsevier, 1999.
- [Goodrich07] Goodrich, Michael and Schultz, Alan.
Human-Robot Interaction: A survey.
Foundations and Trends in Human-Robot Interaction.
vol. 1, no. 3, pp. 203-275.
2007.
- [Guha13] Guha, Anupam and Yang, Yezhou and Fermuuller, Cornelia and Aloimonos, Yiannis.
Minimalist plans for interpreting manipulation actions.
International Conference on Intelligent Robots and Systems (IROS).
pp. 5908–5914.
IEEE, 2013.

- [Harnad90] Harnad, Stevan.
The symbol grounding problem.
 Physica D: Nonlinear Phenomena.
 vol. 42, no. 1-3, pp. 335-346.
 Elsevier, 2001.
- [Hjelmas01] Hjelmås, Erik and Low, Boon Kee.
Face detection: A survey.
 Computer Vision and Image Understanding.
 vol. 83, no. 3, pp. 236-274.
 Elsevier, 2001.
- [Hoermann91] Hoermann, Andreas and Rembold, Ulrich.
Development of an advanced robot for autonomous assembly.
 International Conference on Robotics and Automation (ICRA).
 pp. 2452–2457.
 IEEE, 1991.
- [Hwang92] Hwang, Yong and Ahuja, Narendra.
Gross Motion planning — a survey.
 ACM Computing Surveys (CSUR).
 vol. 24, no. 3, pp. 219-291.
 ACM, 1992.
- [Hyvarinen03] Hyvarinen, Irma.
Der verbale Valenzträger.
 Dependenz und Valenz / Dependency and Valency.
 vol. 1, pp. 738-764.
 Walter de Gruyter, 2003.
- [IDS16] Institut für Deutsche Sprache (IDS).
E-VALBU: Das elektronische Valenzwörterbuch deutscher Verben.
<http://hypermedia.ids-mannheim.de/evalbu/index.html>
 (Letzter Aufruf: 17.12.2016). 2016.
- [IFR16] International Federation of Robotics (IFR).
Executive Summary World Robotics 2016 Service Robots.
 Technical Report, 2016.
- [IPA16] Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA).
Care-o-Bot 4 Anwendungsszenarios.
http://www.care-o-bot.de/content/dam/careobot/en/images/Care-0-bot-4/Download/Application%20Scenarios/150114_Wohnnung.jpg
 (Letzter Aufruf: 07.11.2016). 2016.

- [IRP16] Institut für Robotik und Prozessinformatik der TU Braunschweig (IRP).
<http://www.rob.cs.tu-bs.de/>
(Letzter Aufruf: 21.11.2016). 2016.
- [ISO09] International Organization of Standardization (ISO).
ISO 80000-1:2009. Quantities and Units—Part 1: General .
2009.
- [Jia16] Jia, Yunyi and She, Lanbo and Cheng, Yu and Bao, Jiatong and
Chai, Joyce and Xi, Ning.
Program robots manufacturing tasks by natural language instructions.
International Conference on Automation Science and Engineering (CASE).
pp. 633-638.
IEEE, 2016.
- [Jurafsky09] Jurafsky, Daniel and Martin, James.
Speech and Language Processing.
ISBN: 0131873210.
Prentice-Hall, Inc., 2009.
- [Kemke06] Kemke, Christel.
Action representation for natural language interfaces to agent systems.
International Conference on Hybrid Information Technology.
vol. 2, pp. 1–5.
IEEE, 2006.
- [Kemke07] Kemke, Christel.
*From Saying to Doing - Natural Language Interaction with
Artificial Agents and Robots*.
Human-Robot Interaction (Chapter 09).
ISBN 978-3-902613-13-4.
I-Tech Education and Publishing, 2007.
- [Kessel12] Kessel, Katja and Reimann, Sandra.
Basiswissen Deutsche Gegenwartssprache.
ISBN 978-3825236922.
UBT GmbH, 2012.
- [Klein03] Klein, Dan and Manning, Christopher.
Accurate Unlexicalized Parsing.
Proceedings of the 41st Meeting of the Association for
Computational Linguistics.
pp. 423-430.
Association for Computational Linguistics, 2003.

- [Knoll97] Knoll, Alois. and Hildenbrandt, B. and Zhang, Jianwei.
Instructing cooperating assembly robots through situated dialogues in natural language.
 International Conference on Robotics and Automation (ICRA).
 vol. 1, pp. 888-894.
 IEEE, 1997.
- [Kollar10] Kollar, Thomas and Tellex, Stefanie and Roy, Deb and Roy, Nicholas.
Toward understanding natural language directions.
 International Conference on Human-Robot Interaction (HRI).
 pp. 259-266.
 IEEE, 2010.
- [Kollar14] Kollar, Thomas and Tellex, Stefanie and Roy, Deb and Roy, Nicholas.
Grounding verbs of motion in natural language commands to robots.
 Experimental robotics.
 pp. 31-47.
 Springer, 2014.
- [Koller98] Koller, Rudolf and Norbert Kastrup.
Prinziplösungen zur Konstruktion technischer Produkte.
 2. Auflage, ISBN: 978-3-642-63712-4.
 Springer Verlag, 1998.
- [Kress07] Kress-Gazit, Hadas and Fainekos, Georgios and Pappas, George.
From structured english to robot motion.
 International Conference on Intelligent Robots and Systems (IROS).
 vol. 5, pp. 2717-2722.
 IEEE, 2007.
- [Kroeger04] Kroeger, Torsten and Finkemeyer, Bernd and Wahl, Friedrich.
A Task Frame Formalism for Practical Implementations.
 International Conference on Robotics and Automation (ICRA).
 vol. 5, pp. 5218-5223.
 IEEE, 2004.
- [Kyriacou02] Kyriacou, Theodoros and Bugmann, Guido and Lauria, Stanislaw.
Vision-based urban navigation procedures for verbally instructed robots.
 International Conference on Intelligent Robots and Systems (IROS).
 vol. 2, pp. 1326-1331.
 IEEE, 2002.

- [Kyriacou05] Kyriacou, Theocharis and Bugmann, Guido and Lauria, Stanislao.
Vision-based urban navigation procedures for verbally instructed robots.
Robotics and Autonomous Systems (RAS).
vol. 51, no. 1, pp. 69–80.
Elsevier, 2005.
- [Laengle95] Laengle, Thomas and Lueth, Tim and Stopp, Eva and
Herzog, Gerd and Kamstrup, Gjertrud.
KANTRA - A natural language interface for intelligent robots.
Intelligent Autonomous Systems (IAS 4).
pp. 357-364.
1995.
- [Lauria02] Lauria, Stanislao and Bugmann, Guido and
Kyriacou, Theocharis and Klein, Ewan.
Mobile robot programming using natural language.
Robotics and Autonomous Systems (RAS).
vol. 38, no. 3, pp. 171-181.
Elsevier, 2002.
- [Luczak93] Luczak, Holger.
Arbeitswissenschaft.
ISBN: 978-3-540-54636-8.
Springer Verlag Berlin Heidelberg GmbH, 1993
- [Lueth94] Lueth, Tim and Laengle, Thomas and Herzog, Gerd and
Stopp, Eva and Rembold, Ulrich.
*KANTRA - human-machine interaction for intelligent robots
using natural language.*
International Workshop on Robot and Human Communication.
pp. 106–111.
IEEE, 1994.
- [Marcus93] Marcus, Mitchell and Marcinkiewicz, Mary Ann and Santorini, Beatrice.
Building a large annotated corpus of English: The Penn Treebank.
Computational linguistics.
vol. 19, no. 2, pp. 313-330.
MIT Press, 1993.
- [Mason79] Mason, Matthew.
Compliance and Force Control for Computer Controlled Manipulators.
Massachusetts Institute of Technology, 1979.

- [Matuszek13] Matuszek, Cynthia and Herbst, Evan and Zettlemoyer, Luke and Fox, Dieter.
Learning to parse natural language commands to a robot control system.
Experimental robotics.
pp. 403-415.
Springer, 2013.
- [Matuszek15] Matuszek, Cynthia.
Talking to Robots: Learning to Ground Human Language in Perception and Execution.
Dissertation.
<http://hdl.handle.net/1773/27447>.
2015.
- [Microsoft16] Microsoft Corporation.
Digitaler Assistent Cortana.
<https://www.microsoft.com/de-de/mobile/experiences/cortana/>
(Letzter Aufruf: 16.11.2016). 2016.
- [Misra15] Misra, Dipendra and Sung, Jaeyong and Lee, Kevin and Saxena, Ashutosh.
Tell me Dave: Context-sensitive grounding of natural language to manipulation instructions.
The International Journal of Robotics Research.
vol. 35, pp. 281-300.
Sage Publications, 2015.
- [Moeslund01] Moeslund, Thomas and Granum, Erik.
A survey of computer vision-based human motion capture.
Computer Vision and Image Understanding.
vol. 81, no. 3, pp. 231-268.
Elsevier, 2001.
- [Mosemann01] Mosemann, Heiko and Wahl, Erik.
Automatic decomposition of planned assembly sequences into skill primitives.
Transactions on Robotics and Automation.
vol. 17, no. 5, pp. 709-718.
IEEE, 2001.
- [Nilsson84] Nilsson, Nils.
Shakey the Robot.
Technical Report.
DTIC Document, 1984.

- [Nordmann14] Nordmann, Arne and Hochgeschwender, Nico and Wrede, Sebastian.
A survey on domain-specific languages in robotics.
International Conference on Simulation, Modeling, and
Programming for Autonomous Robots.
pp. 195–206.
Springer, 2014.
- [Nof99] Nof, Shimon.
Handbook of Industrial Robotics.
ISBN: 0-471-17783-0.
John Wiley & Sons, 1999.
- [Onnasch16] Ohnnasch, Linda and Maier, Xenia and Jürgensohn, Thomas.
Mensch-Roboter-Interaktion - Eine Taxonomie für alle Anwendungsfälle.
bua: Fokus (Juni 2016).
Bundesanstalt für Arbeitsschutz und Arbeitsmedizin (bua).
DOI: 10.21934/bua:fokus20160630.
2016.
- [Orendt15] Orendt, Eric and Henrich, Dominik.
*Design of Robust Robot Programs:
Deviation Detection and Classification using Entity-based Resources.*
International Conference on Robotics and Biomimetics (ROBIO).
pp. 1704–1710.
IEEE, 2015.
- [Petrov11] Petrov, Slav and Das, Dipanjan and McDonald, Ryan.
A universal part-of-speech tagset.
arXiv:1104.2086.
2011.
- [Piaget52] Piaget, Jean and Cook, Margaret.
The origins of intelligence in children.
vol. 8, no. 5.
International Universities Press New York, 1952.
- [Pires05] Pires, Norberto.
*Robot-by-voice: experiments on commanding an industrial robot
using the human voice.*
Industrial Robot: An International Journal.
vol. 32, no. 6, pp. 505-511.
Emerald Group Publishing Limited, 2005.

- [QT17] The QT Company.
QT Framework.
<https://www.qt.io/>
 (Letzter Aufruf: 30.01.2017). 2017.
- [Rautaray15] Rautaray, Siddharth and Agrawal, Anupam.
Vision based hand gesture recognition for human computer interaction: a survey.
 Artificial Intelligence Review.
 vol 43, no. 1, pp. 1-54.
 Springer, 2015.
- [Roetting05] Rötting, Matthias and Seifert Katharina.
Multimodale Mensch-Roboter-Interaktion.
 Beiträge zur Mensch-Maschine-Systemtechnik.
 pp. 283-300.
 Symposion Verlag, Düsseldorf, 2005.
- [Roth00] Roth, Karlheinz.
Konstruieren mit Konstruktionskatalogen
Band 1: Konstruktionslehre.
 3. Auflage, ISBN: 978-3-642-62099-7.
 Springer Verlag, 2000.
- [Rybski07] Rybski, Paul and Yoon, Kevin and Stolarz, Jeremy and Veloso, Manuela.
Interactive robot task training through dialog and demonstration.
 International conference on Human-Robot Interaction (HRI).
 pp. 49–56.
 ACM, 2007.
- [Rybski08] Rybski, Paul and Stolarz, Jeremy and Yoon, Kevin and Veloso, Manuela.
Using dialog and human observations to dictate tasks to a learning robot assistant.
 Intelligent Service Robotics.
 vol. 1, no. 2, pp. 159-167.
 Springer, 2008.
- [Sammut11] Sammut, Claude and Webb, Geoffrey.
Encyclopedia of machine learning.
 ISBN: 978-0-387-30768-8.
 Springer Science & Business Media, 2011.

- [Sato87] Sato, Tomomasa and Hirai, Shigeoki.
Language-aided robotic teleoperation system for advanced teleoperation.
IEEE Journal on Robotics and Automation.
vol. 3, no. 5, pp. 476-481.
IEEE, 1987.
- [Schumacher04] Schumacher, Helmut.
VALBU-Valenzwörterbuch deutscher Verben.
Gunter Narr Verlag, 2004.
- [Skut97] Skut, Wojciech and Krenn, Brigitte and Brants, Thorsten and
Uszkoreit, Hans.
An annotation scheme for free word order languages.
Applied natural language processing.
pp. 88-95.
Association for Computational Linguistics, 1997.
- [Spangenberg14] **Spangenberg, Michael** and Henrich, Dominik.
*Towards an intuitive interface for instructing robots handling tasks
based on verbalized physical effects.*
International Symposium on Robot and Human
Interactive Communication (RO-MAN).
pp. 79-84.
IEEE, 2014.
- [Spangenberg15] **Spangenberg, Michael** and Henrich, Dominik.
*Grounding of actions based on verbalized physical effects
and manipulation primitives.*
International Conference on Intelligent Robots and Systems (IROS).
pp. 844-851.
IEEE, 2015.
- [Spangenberg16a] **Spangenberg, Michael** and Henrich, Dominik.
Towards a domain specific language for sensor-based actions.
Applied Mechanics & Materials.
Robotics and Automated Production Lines.
vol. 840 pp. 42-49.
Trans Tech Publications, 2016.
- [Spangenberg16b] **Spangenberg, Michael** and Henrich, Dominik.
*Symbol Grounding for symbolic robot commands
based on physical properties.*
International Conference on Information and Automation (ICIA).
pp. 62-68.
IEEE, 2016.

- [Spangenberg16c] **Spangenberg, Michael** and Henrich, Dominik.
Symbolic Robot Commanding Utilizing Physical Properties - System Overview.
 Joint German/Austrian Conference on Artificial Intelligence (KI).
 pp. 192-199.
 Springer, 2016.
- [SPARC15] SPARC Robotics.
Robotics 2020: Multi-Annual Roadmap for Robotics in Europe.
 Technical Report, 2015.
- [SQLite17] The SQLite Consortium.
SQLite Framework.
<https://www.sqlite.org/index.html>
 (Letzter Aufruf: 30.01.2017). 2017.
- [Stenmark15] Stenmark, Maj.
Instructing industrial robots using high-level task descriptions.
 Licentiate Thesis.
 2015.
- [Suarez12] Suarez, Jesus and Murpy, Robin.
Hand gesture recognition with depth images: A review.
 International Symposium on Robot and Human
 Interactive Communication (RO-MAN).
 pp. 411-417.
 IEEE, 2012.
- [Taylor03] Taylor, Ann and Marcus, Mitchell and Santorini, Beatrice.
The Penn treebank: an overview.
 Treebanks.
 pp. 5-22.
 Springer, 2003.
- [Tellex11] Tellex, Stefanie and Kollar, Thomas and Dickerson, Steven and
 Walter, Matthew and Banerjee, Ashis and Teller, Seth and Roy, Nicholas.
*Understanding natural language commands for robotic navigation
 and mobile manipulation.*
 National Conference on Artificial Intelligence (AAAI 2011).
 pp. 1507-1514.
 AAAI, 2011.

- [Tenorth10] Tenorth, Moritz and Nyga, Daniel and Beetz, Michael.
Understanding and executing instructions for everyday manipulation tasks from the world wide web.
International Conference on Robotics and Automation (ICRA).
pp. 1486-1491.
IEEE, 2010.
- [Tenorth11] Tenorth, Moritz and Klank, Ulrich and Pangercic, Dejan and Beetz, Michael.
Web-enabled Robots: Robots that use the Web as an Information Resource.
IEEE Robotics & Automation Magazine.
vol. 18, no. 2, pp. 58-68.
IEEE, 2011.
- [Tesniere80] Tesnière, Lucien.
Grundzüge der strukturalen Syntax.
ISBN: 978-3129117903.
Klett-Cotta, 1980.
- [Thomas03] Thomas, Ulrike and Finkemeyer, Bernd and Kroeger, Torsten and Wahl, Friedrich.
Error-tolerant execution of complex robot tasks based on skill primitives.
International Conference on Robotics and Automation (ICRA).
vol. 3, pp. 3069-3075.
IEEE, 2003.
- [Thomas08] Thomas, Ulrike.
Automatisierte Programmierung von Robotern für Montageaufgaben.
ISBN: 3832271015.
Shaker Verlag, 2008.
- [Thomas12] Thomas, Brian and Jenkins, Odest.
RoboFrameNet: Verb-centric semantics for actions in robot middleware.
International Conference on Robotics and Automation (ICRA).
pp. 4750-4755.
IEEE, 2012.
- [Torrance94] Torrance, Mark.
Natural Communication with Robots.
Master's thesis.
MIT, Department of Electrical Engineering and Computer Science.
Cambridge, 1994.

- [TUM16] Technische Universität München (TUM).
Service-Roboter James in der Küche.
https://portal.mytum.de/pressestelle/meldungen/NewsArticle_20120523_114511/roboter.jpg
 (Letzter Aufruf: 07.11.2016). 2016.
- [VanDeursen00] Van Deursen, Arie and Klint, Paul and Visser, Joost.
Domain-Specific Languages: An Annotated Bibliography.
 Sigplan Notices.
 vol. 35, no. 6, pp. 26-36.
 2000.
- [Vogt02] Vogt, Paul.
The physical symbol grounding problem.
 Cognitive Systems Research.
 vol. 3, no. 3, pp. 429-457.
 Springer, 2002.
- [Voelter13] Voelter, Markus and Benz, Sebastian and Dietrich, Christian and Engelmann, Birgit and Helander, Mats and Kats, Lennart and Visser, Eelco and Wachsmuth, Guido.
DSL Engineering - Designing, Implementing and Using Domain-Specific Languages.
 ISBN: 978-1-4812-1858-0.
 CreateSpace Independent Publishing Platform, 2013.
- [Vuillaume03] Vuillaume, Marcel.
Valenz und Satzbauplan.
 Dependenz und Valenz/Dependency and Valency:
 Ein Internationales Handbuch Der Zeitgenössischen Forschung/
 An International Handbook of Contemporary Research.
 vol. 1, pp. 738-764.
 Walter de Gruyter, 2003.
- [Werner16] Werner, Tobias and Gradmann, Michael and Orendt, Eric and Sand, Maximilian and **Spangenberg, Michael** and Henrich, Dominik.
ENACT: An efficient and extensible entity-actor framework for modular robotics software components.
 International Symposium on Robotics (ISR).
 pp. 1-7.
 VDE Verlag GmbH, 2016.

- [White40] White, Leslie.
The symbol: The origin and basis of human behavior.
Philosophy of Science.
vol. 7, no. 4, pp. 451-463.
1940.
- [Yanco04] Yanco, Holly and Drury, Jill.
Classifying Human-Robot Interaction: An Updated Taxonomy.
International Conference on Systems, Man and Cybernetics.
vol. 3, pp. 2841-2846.
IEEE, 2004.
- [Yang14] Yang, Yezhou and Guha, Anupam and Fermüller, Cornelia and Aloimonos, Yiannis.
Manipulation action tree bank: A knowledge resource for humanoids.
International Conference on Humanoid Robots.
pp. 987-992.
IEEE, 2014.
- [Zampogiannis15] Zampogiannis, Konstantinos and Yang, Yezhou and Fermüller, Cornelia and Aloimonos, Yiannis.
Learning the spatial semantics of manipulation actions through preposition grounding.
International Conference on Robotics and Automation (ICRA).
pp. 1389–1396.
IEEE, 2015.
- [Zhang99] Zhang, Jianwei and Von Collani, Yorck and Knoll, Alois.
Interactive assembly by a two-arm robot agent.
Robotics and Autonomous Systems (RAS).
vol. 29, no. 1, pp. 91–100.
Elsevier, 1999.
- [Zhao03] Zhao, Wenyi and Chellappa, Rama and Phillips, Jonathon and Rosenfeld, Azriel.
Face recognition: A literature survey.
ACM computing surveys (CSUR).
vol. 35, no. 4, pp. 399-458.
ACM, 2003.
- [Zielinski95] Zieliński, Cezary.
Robot programming methods.
ISSN: 0137-2319.
Publishing House of Warsaw University of Technology.
1995.

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die von mir angegebenen Quellen und Hilfsmittel verwendet habe.

Weiterhin erkläre ich, dass ich die Hilfe von gewerblichen Promotionsberatern bzw. –vermittlern oder ähnlichen Dienstleistern weder bisher in Anspruch genommen habe, noch künftig in Anspruch nehmen werde.

Zusätzlich erkläre ich hiermit, dass ich keinerlei frühere Promotionsversuche unternommen habe.

Bayreuth, den

Michael Spangenberg